Best S-box amongst differently sized S-boxes based on the avalanche effect in the advance encryption standard algorithm

Hadeel Mohammed Taher[1], Seddiq Qais Abd Al-Rahman[2], Shihab A. Shawkat[3]
[1]Department of Quality Assurance and Academic Performance, University of Anbar, Ramadi, Iraq
[2]Department of Computer Networks Systems, College of Computer Science and Information Technology, University of Anbar, Ramadi, Iraq
[3]Department of Quality Assurance and Academic Performance, University Presidency, University of Samarra, Samarra, Iraq

Article Info

Article history:

Received Aug 9, 2021
Revised Jul 13, 2022
Accepted Aug 1, 2022

Keywords:

Advanced encryption standard
Avalanche effect
Cryptography algorithm
Information security
S-box

ABSTRACT

Substitution boxes are essential nonlinear modules that are popular in block cipher algorithms. They also play a significant role in the security area because of their robustness to different linear cryptanalysis. Each element of the state in a S-box is nonlinearly replaced using a lookup table. This research presents the S-box, one of the fundamental parts of the advanced encryption standard (AES) algorithm. The S-box represents the confusion part in the AES. However, when information is shared between different devices in an authorized manner, the algorithm should be able to combine a sufficient number of confusion layers to guarantee the avalanche effect (AE). Subsequently, this research selects the best S-box by comparing different sizes (4×4, 8×8, and 16×16) and measuring them on the basis of the million-bit encryption. The AE is the main criterion used in choosing the best S-box. A robust and strong cryptography algorithm should be able to confirm the AEs. Results indicate that the 16×16 S-box with a 52% AE ratio is the superior S-box.

Corresponding Author:

Seddiq Qais Abd Al-Rahman
Department of computer networks systems, College Computer Science and Information Technology, University of Anbar
Ramadi, Iraq
Email: co.sedeikaldossary@uoanbar.edu.iq

1. INTRODUCTION

With the speedy evolution of digital documents and data exchange, the process of protecting data also needs to be advanced to ensure the safety of data from loss or modification [1]. Cryptographic algorithms can be classified according to symmetrical and asymmetrical encryption algorithms [2]. Several cryptographic algorithms have been proposed, and the most popular amongst them are the advanced encryption standard (AES) and data encryption standard (DES) algorithms. DES, initially developed by International Business Machines Corporation (IBM), initiates the encryption by using a 64-bit key [3]. In succeeding years, DES was improved as a component function by means of triple DES [4]. However, the IBM in United States of America stopped supporting DES in 1998 when a supercomputer along with other distributed computers cracked this algorithm in only 22 h [5]. In 2001, the National Institute of Standards and Technology introduced the AES [6], an algorithm designed by a group of researchers who could resolve the DES cracking issue [7]. Nonetheless, the cryptographic algorithms of AES and DES must be explained and examined. The substitution box (S-box) of the block cipher an important tool in public key cryptography was created in Lucifer using a 4-bit Boolean function for encryption and decryption algorithms in the late 1970s. The S-box of the AES algorithm, created using a polynomial for Galois field (GF) (28) calculation, has been continuously applied since the early 20th century. Three kinds of S-boxes are currently used, and each of them differs from the other in terms of size. The S-box is a box corresponding to a number of elements (i.e. 4 bits, 24=16).

The values can be varied from 0 to F via hexadecimal representation and organized in arbitrary mode. For instance, 4 bits in DES can be represented as a 16-bit output vector, in which each bit is an output bit equivalent to 16 possibilities of the 4-bit sequential input, starting with ‘0000’ and ending with ‘1111’. Thus, the single 4-bit function is carried out in 16 rounds in parallel. As for an S-box with 8 bits, the box of elements is given by 28=256. The values can be varied from 0 to 255, similar to that used in AES. The 8-bit function, which gives a 1-bit output for 8 input bits, ultimately takes the form of a 256-bit column vector. Thus, 8 bits correspond to a 256-bit output vector, in which each bit is an output bit equivalent to 256 possibilities of the 8-bit sequential input, starting with ‘00000000’ and ending with ‘11111111’. When 16 bits are used as the output in the S-box, the possibilities equal 216=65536. The decimal values differ from 0 to 65535 [8].

Past studies have explored the development of the S-box scheme algorithm by using chaotic systems in different approaches. Shawkat *et al.* [8] developed an S-box scheme algorithm and a block encryption procedure by using a chaotic map and a logistic map for the cryptanalysis. Banerjee *et al.* [9] recommended the use of an algorithm with image encryption by using the Chen chaotic map with the chaos-based S-box. Shawkat *et al.* [10] designed a novel S-box generation procedure based on the scaled Zhong chaotic system. A new random number generator based on the originally scaled Zhongtang chaotic concept combined with highly complex and dynamic properties were considered. Their S-box, which was compared with those in other studies, was more effective and stronger. Renuka *et al.* [11] evaluated an S-box and compared it using typical arithmetic tests under similar bijective characteristics. Results from the strict avalanche (SAC), nonlinearity and equiprobable input/output XOR spreading tests obtained good results. Moreover, their S-box was compared with several recent chaos-constructed S-boxes. Further results proved that their scheme was consistent and suitable for protected communication. Altigani *et al.* [12] implemented cryptanalysis for an image encryption procedure by creating an S-box based on the chaos principle. Their system offered the benefits of simple organization, good encryption performance and high encryption effectiveness.

In this paper, we explain the AES algorithm and compare differently sized S-boxes to determine which one amongst them has a stronger avalanche effect (AE). The geometry and full dimensions of these projectile shapes are shown in Figure 1. The models are: i) cone-cylinder, ii) ogive-cylinder, iii) blunted cone-cylinder, iv) cone-cylinder boattail (4°), and v) cone-cylinder boattail (8°). All the models have a fineness ratio of 6.67 and a center-of-gravity location at about the 40% body station. The supersonic Mach number range considered is from 1.6 to 5 for zero-angle of attack.

2. ADVANCED ENCRYPTION STANDARD INTERNAL STRUCTURE

The AES algorithm is currently the strongest published symmetric cryptographic algorithm because it uses the identical (same key) secret key for encryption and decryption. The size of the key for any encryption algorithm is responsible for the strength of that algorithm [8]. The AES algorithm is a flexible method because it comprises three key sizes (128, 192 and 256 bits) with 128-bit block-size data. When the cipher text is used with the plain text of the AES algorithm, the operation must iterate a certain number of times [9]. The key size is used to calculate the number of rounds from which the plain text is handled [10]. The details of the AES rounds with a 128-bit block size are shown in Table 1.

Table 1. AES key length with different numbers of rounds

Key Length	Number of Round
128	10
192	12
256	14

The AES algorithm is robust according to the criteria set defined by the National Institute of Standards and Technology Framework (NIST) cybersecurity. The three main standards considered are as [11]:

- Security (confidentiality): the core emphasis and the core attention of the AES algorithm is security. The data are protected from attacks, including cryptanalysis attacks (e.g. brute force).
- Cost: the AES algorithm contributes to computational efficiency whilst maintaining the usual processes for site implementation (e.g. software and hardware).
- Implementation: the AES algorithm is flexible as it can be executed on any platform [2].

However, the following layer categories need to be addressed in the AES procedure:

- The key layer simply implements the XOR operation between the working state with the round key.
- The confusion layer is an essential element in cryptographic processes. The confusion layer is handled in the nonlinear section by using the byte substitution layer (i.e. the S-box is a charting table that transforms the n-bit into m-bits). The confusion work should be able to substitute the contented state table of the plain text with the supplementary contented S-box.
- The diffusion layer is responsible for diffusing all of the above state bits. This layer has two sub-diffusion layers. The first sub-diffusion layer called the shift row uses the technique of shifting the rows of the overhead layer output. The second sub-diffusion layer called the mixed column is a matrix with continuous entries, each with a 4-byte column represented as a vector, which is then multiplied in a fixed 4×4 matrix [12].

Figure 1 shows the details of the AES procedure and the sequence of transfers in each round. The AES performs an extra final routine that is composed of (*SubBytes*, *ShiftRows* and *AddRoundKey*) steps, The input to the encryption and decryption algorithms is a single 128-bit block. this block is depicted as a square matrix of bytes. This block is copied into the State array, which is modified at each stage of encryption or decryption. After the final stage, State is copied to an output matrix.

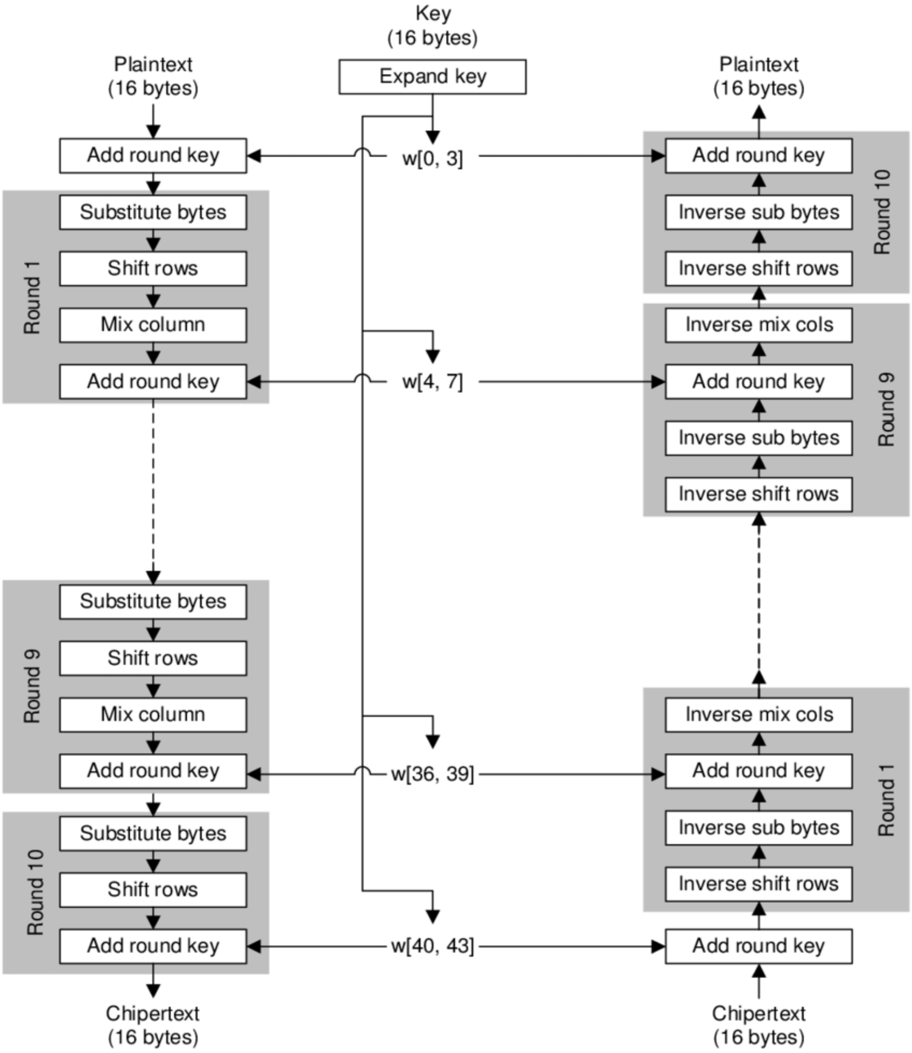


Figure 1. AES procedure

The confusion and diffusion layers are significant characteristics of the AES procedure because they are intended to address the information security issue. The simplest technique for handling in both diffusion and confusion in the cryptographic process is to use a substitution and permutation function. In this case, the most challenging task in ensuring the strength of the AES algorithm in cryptanalysis is securing the S-box [13]. Thus, the design and properties of the S-box should be fully understood prior to its application in the AES algorithm. The five properties of the S-box are as [14]:

- Avalanche effect: AE is an important characteristic of the AES algorithm. The term ‘avalanche effect’ was initially used via Horst Feistel in 1973 [15].
- Strict avalanche: SAC is delineated by entirely changing the output bits based on a partial probability when only a 1-bit input is completed [5].
- Bit independence: Initially defined as a concept by Webster and Tavares [4], the bit independent criterion is currently used as a set of specific avalanche vectors. This criterion relies on the pairwise self-governance of avalanche variables.
- Nonlinearity: This property is imperative and the most important amongst all cryptographic properties. A robust cryptographic method requires high nonlinearity. The resistance of a method is calculated using a set of linear equations; then, the resistance is confirmed in relation to the linear cryptanalysis [6].

The main and most important properties of the S-box are related to AE. In this study, we explore the features of AE and compare differently sized S-boxes to determine which amongst them has the strongest AE for the AES algorithm. The performance of AES according to different S-box sizes is evaluated using different measurements. Many experiments have been applied, and all the results have been reported in tables in the experiment sections.

3. AVALANCHE EFFECT

The AE is an important parameter for measuring the strength of algorithms. However, other parameters can be used to measure the strength of an algorithm, including the ones mentioned in section 1 (introduction). As the AE is one of the most efficient parameters for measuring the strength of the AES algorithm, the AE aspect also needs to be comprehended.

The AE is an important characteristic of the AES algorithm. The term ‘avalanche effect’ was first used by Horst Feistel in 1973 [15]. Then, the AE was conceptualized using the ‘Shannon’s property of confusion’ [16]. For example, one of the principal design objectives of an algorithm is the implementation of a robust cipher or cryptography hash function [4]. When the algorithm requires a hash value, even a small change in the plain text input string will considerably alter the hash value. Consequently, the property after altering only 1 bit in the plain text may be changed; in such a case, any modified result can be inspected as a minimum-half of the bits in the output cipher text [2].

One of the goals of AE is to change only 1 bit, but this small change can considerably alter the final result. In such a situation, the cipher text cannot be easily analyzed whilst attempting to determine a threat. The AE in the AES algorithm and the S-box can be calculated by altering 1 bit in the plain text, i.e. from ‘11’ to ‘12’ or from ‘00’ to ‘10’ [8]. The AE is computed as:

$$Avalanche\ Effect\ = \frac{number\ of\ flipped\ bits\ in\ cipher\ text}{number\ of\ bits\ in\ cipher\ text}$$

A slight change in the bits in the plain text can cause an ‘avalanche’ of modifications. This situation means that the outcomes will require have a different large number of the cipher text. The function $f\colon \{\,0{,}1\}^n \to \{0{,}1\}^n$ must satisfy the avalanche criterion if at any time 1 bit is changed in the input. Usually, the half-bits in the output are changed. In the equations, i and $j \in (1, 2, \ldots, n)$ are the bits representing the input with the output. The $n \times n$ S-box in this study should be able to satisfy the avalanche criteria only in the case where each $i = 1, 2, \ldots, n$;

$$\frac{1}{2^n}\sum_{j=1}^{n} W(a_j^{ei}) = \frac{n}{2} \quad (1)$$

$$W\left(a_j^{ei}\right) = \sum_{all\ X\in\{0,1\}^n} a_j^{ei} \quad (2)$$

where a_j^{ei} is the entire modification, in which j' is differently valued for the AE. Then, a_j^{ei} is calculated to complete the entire plain text input alphabet of the 2nd dimension. Here, $0 \leq W(a_j^{ei}\ ) \leq 2n.$ As shown in (2) can then be used to describe the AE parameter. In particular, $kAE(i)$ is defined as (3),

$$kAE(i) = \frac{1}{n2^n}\sum_{j=1}^{n} W(a_j^{ei}) = \frac{1}{2} \quad (3)$$

where $kAE(i)$ takes a value in the range of [0, 1]. The range can be interpreted as the possibility of modification for all bits in the cipher text after only the i^{-th} bit in the input plain text is altered. If $kAE(i)$ is not similar to 1/2 for any i, then the S-box does not satisfy the AV measure [14]. In ensuring good encryption, the AES algorithm should always satisfy the following relation [16], [17].

$$Avalanche\ effect\ >\ 50\%.$$

4. PROPOSED METHOD

4.1. Experimental design

In this study, the AES algorithm was executed using Python. A personal computer powered by Core i7 6400U, 8 GB RAM and 500 GB of hard disk drive storage capacity was used. The performance of the AES algorithm was evaluated by calculating the AE parameter. A randomness test was carried out for the NIST statistical tests, and the execution time was calculated.

4.2. Results and analyses

The suggested approach entailed the use of the features of the AES cryptographic algorithm. The bit sizes for the encryption were as: 128 bits, 1,024 bits and 1,048,576 bits (1 megabit). The message (plain text) to be encrypted was split into blocks with 64 bits. Each block was enciphered using the Play-Fair cipher text. Then, the encrypted message was further processed by intensive scrambling, and the scrambled message had 8 bits. The AES structure and its round function procedure can be summarized as:

- Add round key: sub-keys of the same size are added to the steps. In all rounds, the sub-key was derived from the central key by using the Rijndael schedule key. The sub-key was added by merging each byte of the step and using the corresponding byte of the sub-key along with the XOR.
- S-box layer: the substitution in the AES block was implemented by the S-box in a byte-by-byte manner.
- Mix-columns: substitution was achieved using an arithmetic operation.
- Mix-rows: the diffusion matrix M was used, and the rows with the 4×4 array were multiplied by the matrix M.

For the AES structure, we considered the approaches proposed in the literature to construct the chaos-based S-boxes with different sizes, namely, the 4×4 [18]–[22], 8×8 [23]–[27] and 16×16 [28]–[32] S-boxes, as shown in Tables 2 to 10. According to reference [18]–[27], [28]–[32] the small pathway delay with a small gate region can be implemented given its low energy. The following verification techniques were used to evaluate the differently sized S-boxes in the AES:

- Time execution: the number of bits encrypted over time is computed, as shown in Tables 2, 5, and 8.
- NIST statistical suite tests: this statistical package consists of 15 tests developed by NIST for testing the randomness of binary but randomly long sequences as produced by the software or based on random coding or pseudo-random number creators. The tests concentrate on a variety of non-randomness categories that may occur in the sequence. Moreover, the tests are decomposable and can be add to several subtests. A number of files encrypted by the AES algorithm are used as the input parameters, then they are subject to the 15 NIST tests. The statistical test results for the different S-box sizes are listed in Tables 3, 6 and 9.
- Avalanche effect: as explained previously, the AE criterion can be used to calculate one million bits. In the experiment, keys are adopted to encrypt a permanent plain text by using the AES algorithm. Tables 4, 7 and 10 show the results of the differently sized S-boxes that passed the AE test.

Table 2. Time execution of the 4×4 S-box

S-Box	No. of bits Encryption	Time (ms)
Bogdanov *et al.* [18]	128	0.32411
	1024	0.80526
	1048576 (1 Megabits)	4.92064
Li *et al.* [19]	128	0.71697
	1024	1.20864
	1048576 (1 Megabits)	6.31325
Shawkat *et al.* [20]	128	0.56040
	1024	0.98308
	1048576 (1 Megabits)	5.71323
Suzaki *et al.* [21]	128	0.46116
	1024	0.87056
	1048576 (1 Megabits)	5.01729
Banik *et al.* [22]	128	0.39747
	1024	0.69936
	1048576 (1 Megabits)	4.26742

Table 3. Randomness test (NIST statistical package) of the 4×4 S-box

Test Name		Bogdanov *et al.* [18]	Li *et al.* [19]	Shawkat *et al.* [20]	Shawkat *et al.* [20]	Banik *et al.* [22]
Frequency test		0.395	0.237	0.198	0.866	0.514
Frequency test within a block		0.168	0.383	0.277	0.089	0.853
Runs test		0.241	0.525	0.313	0.840	0.088
The longest run of ones in a block		0.624	0.603	0.511	0.123	0.993
Cumulative sums test	REVE-RSE	0.631	0.103	0.795	0.879	0.395
	FOR-WARD	0.325	0.515	0.312	0.927	0.525
Serial test	P-v1	0.733	0.884	0.382	0.864	0.375
	P-v2	0.572	0.619	0.377	0.975	0.825
Approximate entropy test		0.246	0.304	0.993	0.745	0.909
Binary matrix rank test		0.493	0.898	0.589	0.016	0.378
Maurer’s ‘universal statistical’ test		0.974	0.409	0.695	0.704	0.463
Non-overlapping template matching test		0.916	0.282	0.835	0.098	0.818
Overlapping template matching test		0.885	0.221	0.894	0.312	0.652
Lempel–Ziv compression test		0.409	0.705	0.450	0.975	0.955
Linear complexity test		0.835	0.145	0.538	0.284	0.419
Random excursion test		0.884	0.205	0.632	0.365	0.325
Random excursion variant test		0.684	0.534	0.845	0.346	0.639

Table 4. AE results of the 4×4 S-box

S-Box	Ratio of avalanche effect
Bogdanov *et al.* [18]	47%
Li *et al.* [19]	38%
Shawkat *et al.* [20]	30%
Suzaki *et al.* [21]	51%
Banik *et al.* [22]	41%

Table 5. Time execution of the 8×8 S-box

S-Box	No. of bits Encryption	Time (ms)
Tran *et al.* [23]	128	0.8038
	1024	1.1074
	1048576 (1 Megabits)	4.9342
Çavuşoğlu *et al.* [24]	128	1.0523
	1024	1.3817
	1048576 (1 Megabits)	6.5109
Liu *et al.* [25]	128	0.6668
	1024	1.3768
	1048576 (1 Megabits)	5.7424
Ahmad *et al.* [26]	128	0.8922
	1024	1.0807
	1048576 (1 Megabits)	5.4636
Lambić [27]	128	0.7935
	1024	0.9302
	1048576 (1 Megabits)	4.4976

Table 6. Randomness test (NIST statistical package) of the 8×8 S-box

Test Name		Tran *et al.* [23]	Çavuşoğlu *et al.* [24]	Liu *et al.* [25]	Ahmad *et al.* [26]	Lambić *et al.* [27]
Frequency test		0.376	0.218	0.125	0.324	0.279
Frequency test within a block		0.985	0.203	0.386	0.468	0.238
Runs test		0.928	0.212	0.315	0.473	0.458
The longest run of ones in a block		0.778	0.757	0.290	0.759	0.332
Cumulative sums test	REVE-RSE	0.062	0.127	0.409	0.335	0.574
	FOR-WARD	0.494	0.368	0.361	0.374	0.355
Serial test	P-v1	0.173	0.168	0.204	0.782	0.181
	P-v2	0.024	0.876	0.379	0.627	0.770
Approximate entropy test		0.728	0.786	0.949	0.905	0.895
Maurer’s ‘universal statistical’ test		0.224	0.659	0.419	0.136	0.923
Binary matrix rank test		0.139	0.544	0.936	0.386	0.446
Non-overlapping template matching test		0.573	0.939	0.226	0.626	0.791
Lempel–Ziv compression test		0.774	0.970	0.663	0.235	0.888
Linear complexity test		0.264	0.574	0.674	0.785	0.454
Random excursion test		0.885	0.206	0.983	0.947	0.617
Overlapping template matching test		0.293	0.629	0.324	0.743	0.117
Random excursion variant test		0.455	0.350	0.545	0.813	0.523

Table 7. AE results of the 8×8 S-box

S-Box	Ratio of avalanche effect
Tran *et al.* [23]	39%
Çavuşoğlu *et al.* [24]	34%
Liu *et al.* [25]	43%
Ahmad *et al.* [26]	37%
Lambić *et al.* [27]	35%

Table 8. Time execution of 16×16 S-box

S-Box	No. of bits Encryption	Time (ms)
Srividya *et al.* [28]	128	1.3252
	1024	2.0615
	1048576 (1 Megabits)	5.9071
Hameed *et al.* [29]	128	2.1687
	1024	1.9057
	1048576 (1 Megabits)	6.7379
Farah *et al.* [30]	128	0.7105
	1024	2.6192
	1048576 (1 Megabits)	6.7302
Belazi *et al.* [31]	128	1.6414
	1024	2.2137
	1048576 (1 Megabits)	5.7315
Zhu *et al.* [32]	128	1.1238
	1024	1.8651
	1048576 (1 Megabits)	5.2329

Table 9. Randomness test (NIST statistical package) of the 16×16 S-box

Test Name		Srividya *et al.* [28]	Hameed *et al.* [29]	Farah *et al.* [30]	Belazi *et al.* [31]	Zhu *et al.* [32]
Frequency test		0.319	0.161	0.272	0.763	0.987
Frequency test within a block		0.556	0.776	0.913	0.982	0.714
Runs test		0.659	0.859	0.595	0.986	0.379
The longest run of ones in a block		0.472	0.472	0.377	0.457	0.689
Cumulative sums test	REVE-RSE	0.702	0.767	0.467	0.336	0.372
	FOR-WARD	0.778	0.697	0.653	0.371	0.978
Serial test	P-v1	0.654	0.773	0.685	0.783	0.654
	P-v2	0.648	0.752	0.816	0.641	0.748
Approximate entropy test		0.341	0.911	0.803	0.244	0.621
Maurer's 'universal statistical' test		0.383	0.843	0.565	0.594	0.912
Binary matrix rank test		0.392	0.792	0.531	0.489	0.993
Non-overlapping template matching test		0.531	0.931	0.914	0.444	0.979
Lempel–Ziv compression test		0.856	0.516	0.814	0.519	0.760
Linear complexity test		0.976	0.546	0.786	0.992	0.817
Overlapping template matching test		0.761	0.361	0.551	0.687	0.643
Random excursion test		0.649	0.975	0.162	0.648	0.590
Random excursion variant test		0.690	0.571	0.502	0.657	0.742

First, the 4×4 S-box was evaluated in terms of time execution and randomness by using the NIST statistical suite. The best S-box would depend on the modified result involving the minimum-half of the bits in the output cipher text. The main criterion used by Suzaki *et al.* [21], i.e. a good ratio for encryption results in more than 50% AE, was adopted. Table 4 shows the AE results of the 4×4 S-box.

Second, the 8×8 S-box was analysed using the same method discussed above for the 4×4 S-box. The 8×8 S-box was evaluated in terms of time execution and randomness by using the NIST statistical suite. The main criterion used by Liu *et al.* [25], i.e. an AE of 43%, was adopted for the 8×8 S-box because its AE is close to 50% AE. Table 7 shows the AE results of the 8×8 S-box.

Finally, the 16×16 S-box was evaluated using the same parameters of the 4×4 and 8×8 S-boxes. The 16×16 S-box was evaluated in terms of time execution and randomness by using the NIST statistical suite. The main criterion used by Belazi *et al.* [31], i.e. an AE of 52%, was adopted for the 16×16 S-box because its AE is close to 50%. A comparison of the S-boxes of Srividya *et al.* [28], Hameed *et al.* [29] and Zhu *et al.* [32] shows that the ratios are close to each other for the avalanche criteria in Table 10. However, distinct features can be inferred when the differently sized S-boxes are compared for the avalanche standard as shown in Table 11. More than half of the bits were changed after 16 rounds of AES. The 16×16 S-box with 52% AE (i.e. Belazi *et al.* [31]) was the best one in this study.

Table 10. AE results of the 16×16 S-box

S-Box	Ratio of avalanche effect
Srividya *et al.* [28]	47%
Hameed *et al.* [29]	49%
Farah *et al.* [30]	37%
Belazi *et al.* [31]	52%
Zhu *et al.* [32]	43%

Table 11. Comparison of AE results of the 4×4, 8×8 and 16×16 S-boxes

Size S-Box	Methods	Ratio of avalanche effect
4×4	Bogdanov *et al.* [18]	47%
	Li *et al.* [19]	38%
	Shawkat *et al.* [20]	30%
	Shawkat *et al.* [20]	51%
	Banik *et al.* [22]	41%
8×8	Tran *et al.* [23]	39%
	Çavuşoğlu *et al.* [24]	34%
	Liu *et al.* [25]	43%
	Ahmad *et al.* [26]	37%
	Lambić [27]	35%
16×16	Srividya *et al.* [28]	47%
	Hameed *et al.* [29]	49%
	Farah *et al.* [30]	37%
	Belazi *et al.* [31]	52%
	Zhu *et al.* [32]	43%

5. CONCLUSION

Encryption algorithms play an essential role in communication security when the level of encryption is the main matter of concern. One of the most commonly used encryption algorithms for performance estimation is the AES. This study demonstrated that the S-box and AE are two important elements of the AES algorithm. In this study, the AE was set high, and different S-box sizes (4×4, 8×8 and 16×16) were used for the substitution. Many researchers are currently interested in knowing the recovery features of the S-box. This research contributes to the knowledge gap by determining the best S-box. Our results showed that the 16×16 S-box with 52% AE is superior to the 4×4 and 8×8 S-boxes. In the future, experimentations can be conducted on images in the AES algorithm. Increasing the security level is another possible focus of future research.

REFERENCES

[1] A. M. Al-Smadi, A. Al-Smadi, R. M. Ali Aloglah, N. Abu-darwish, and A. Abugabah, "Files cryptography based on one-time pad algorithm," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 3, pp. 2335–2342, Jun. 2021, doi: 10.11591/ijece.v11i3.pp2335-2342.

[2] A. F. Shimal, B. H. Helal, and A. T. Hashim, "Extended of TEA: a 256 bits block cipher algorithm for image encryption," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 5, pp. 3996–4007, Oct. 2021, doi: 10.11591/ijece.v11i5.pp3996-4007.

[3] A. K. Mandal, C. Parakash, and A. Tiwari, "Performance evaluation of cryptographic algorithms: DES and AES," in *IEEE Students' Conference on Electrical, Electronics and Computer Science*, 2012, pp. 1–5, doi:: 10.1109/SCEECS.2012.6184991.

[4] M. Khan and T. Shah, "A construction of novel chaos base nonlinear component of block cipher," *Nonlinear Dynamics*, vol. 76, no. 1, pp. 377–382, 2014, doi: 10.1007/s11071-013-1132-0.

[5] M. Ahmad, M. N. Doja, and M. M. S. Beg, "ABC optimization based construction of strong substitution-boxes," *Wireless Personal Communications*, vol. 101, no. 3, pp. 1715–1729, Aug. 2018, doi: 10.1007/s11277-018-5787-1.

[6] T. Manoj Kumar and P. Karthigaikumar, "A novel method of improvement in advanced encryption standard algorithm with dynamic shift rows, sub byte and mixcolumn operations for the secure communication," *International Journal of Information Technology*, vol. 12, no. 3, pp. 825–830, Sep. 2020, doi: 10.1007/s41870-020-00465-1.

[7] C. P. Dewangan, S. Agrawal, A. K. Mandal, and A. Tiwari, "Study of avalanche effect in AES using binary codes," in *IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, Aug. 2012, pp. 183–187, doi: 10.1109/ICACCCT.2012.6320767.

[8] S. A. Shawkat, O. Abu-Elnasr, and T. Elarif, "Evolved algorithm to secure communication with steganography," *International Journal of Intelligent Computing and Information Science (IJICIS)*, vol. 17, no. 1, pp. 1–17, 2017. doi:10.1109/researchgate.net/publication/314307858.

[9] B. Banerjee, "Avalanche effect : a judgement parameter of strength in symmetric key block ciphers," *International Journal of*

Engineering Development and Research, vol. 7, no. 2, pp. 116–121, 2019.

[10] S. A. Shawkat, “Enhancing steganography techniques in digital images,” Faculty of Computers and Information, Mansoura University, 2016, doi: 10.13140/RG.2.2.16678.57925.

[11] G. Renuka, V. U. Shree, and P. C. S. Reddy, “Comparison of AES and des algorithms implemented on virtex-6 FPGA and Microblaze soft core processor,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 5, pp. 3544–3549, Oct. 2018, doi: 10.11591/ijece.v8i5.pp3544-3549.

[12] 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.

[13] A. Ahlawat and V. Nandal, “Ruggedizing LTE security using hybridization of AES and RSA to provide double layer security,” in *Lecture Notes on Data Engineering and Communications Technologies*, vol. 54, 2021, pp. 121–135, doi: 10.1007/978-981-15-8335-3_12.

[14] I. Vergili and M. D. Yücel, “Avalanche and bit independence properties for the ensembles of randomly chosen n×n s-boxes,” *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 9, no. 2, pp. 137–145, 2001.

[15] J. C. Hernandez-Castro, J. M. Estevez-Tapiador, A. Ribagorda-Garnacho, and B. Ramos-Alvarez, “Wheedham: an automatically designed block cipher by means of genetic programming,” *2006 IEEE Congress on Evolutionary Computation, CEC 2006*, pp. 192–199, 2006, doi: 10.1109/cec.2006.1688308.

[16] L. R and K. M, “Enhancing the security of AES through small scale confusion operations for data communication,” *Microprocessors and Microsystems*, vol. 75, Jun. 2020, doi: 10.1016/j.micpro.2020.103041.

[17] P. Witoolkollachit, “The avalanche effect of various hash functions between encrypted raw images versus non-encrypted images: a comparison study,” *Journal of the Thai Medical Informatics Association*, vol. 1, pp. 69–82, 2016.

[18] A. Bogdanov *et al.*, “PRESENT: an ultra-lightweight block cipher,” in *Cryptographic Hardware and Embedded Systems*, vol. 4727, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 450–466, doi: 10.1007/978-3-540-74735-2_31.

[19] L. Li, B. Liu, Y. Zhou, and Y. Zou, “SFN: a new lightweight block cipher,” *Microprocessors and Microsystems*, vol. 60, pp. 138–150, Jul. 2018, doi: 10.1016/j.micpro.2018.04.009.

[20] S. A. Shawkat, K. S. L. Al-badri, and A. Ibrahim Turki, “The new hand geometry system and automatic identification,” *Periodicals of Engineering and Natural Sciences (PEN)*, vol. 7, no. 3, Aug. 2019, doi: 10.21533/pen.v7i3.632.

[21] T. Suzaki, K. Minematsu, S. Morioka, and E. Kobayashi, “TWINE: a lightweight, versatile block cipher,” in *ECRYPT Workshop on Lightweight Cryptography*, 2011, pp. 146–169.

[22] S. Banik *et al.*, “WARP : revisiting GFN for lightweight 128-bit block cipher,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12804, Springer International Publishing, 2021, pp. 535–564, doi: 10.1007/978-3-030-81652-0_21.

[23] M. T. Tran, D. K. Bui, and A. D. Duong, “Gray S-Box for advanced encryption standard,” in *International Conference on Computational Intelligence and Security*, Dec. 2008, vol. 1, pp. 253–258, doi: 10.1109/CIS.2008.205.

[24] Ü. Çavuşoğlu, A. Zengin, I. Pehlivan, and S. Kaçar, “A novel approach for strong S-Box generation algorithm design based on chaotic scaled Zhongtang system,” *Nonlinear Dynamics*, vol. 87, no. 2, pp. 1081–1094, 2017, doi: 10.1007/s11071-016-3099-0.

[25] J. Liu, B. Wei, X. heng, and X. Wang, “An AES S-box to increase complexity and cryptographic analysis,” in *19th International Conference on Advanced Information Networking and Applications*, 2005, vol. 1, pp. 724–728, doi: 10.1109/AINA.2005.84.

[26] M. Ahmad, P. M. Khan, and M. Z. Ansari, “A simple and efficient key-dependent S-box design using fisher-yates shuffle technique,” in *Communications in Computer and Information Science*, vol. 420 CCIS, 2014, pp. 540–550, doi: 10.1007/978-3-642-54525-2_48.

[27] D. Lambić, “A new discrete-space chaotic map based on the multiplication of integer numbers and its application in S-box design,” *Nonlinear Dynamics*, vol. 100, no. 1, pp. 699–711, Mar. 2020, doi: 10.1007/s11071-020-05503-y.

[28] S. R. and R. B., “Implementation of AES using biometric,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 5, pp. 4266–4276, Oct. 2019, doi: 10.11591/ijece.v9i5.pp4266-4276.

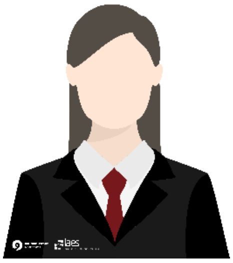
[29] M. E. Hameed, M. Mat Ibrahim, N. Abd Manap, and M. L. Attiah, “Comparative study of several operation modes of AES algorithm for encryption ECG biomedical signal,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 6, pp. 4850–4859, Dec. 2019, doi: 10.11591/ijece.v9i6.pp4850-4859.

[30] T. Farah, R. Rhouma, and S. Belghith, “A novel method for designing S-box based on chaotic map and teaching–learning-based optimization,” *Nonlinear Dynamics*, vol. 88, no. 2, pp. 1059–1074, Apr. 2017, doi: 10.1007/s11071-016-3295-y.

[31] A. Belazi and A. A. A. El-Latif, “A simple yet efficient S-box method based on chaotic sine map,” *Optik*, vol. 130, pp. 1438–1444, Feb. 2017, doi: 10.1016/j.ijleo.2016.11.152.

[32] C. Zhu, G. Wang, and K. Sun, “Cryptanalysis and improvement on an image encryption algorithm design using a novel chaos based s-box,” *Symmetry*, vol. 10, no. 9, Sep. 2018, doi: 10.3390/sym10090399.

BIOGRAPHIES OF AUTHORS



Hadeel Mohammed Taher received the B.Sc. degree in Computer Science in 2010 and M.Sc. Degree with distinction in Computer Science in 2013 from University of Anbar/Iraq. She worked as lecture in University of Anbar. The interest research in computer science in general and security, network and AI as exact specialization. She can be contacted at e-mail: hudeel.mohammed@uoanbar.edu.edu.iq.

Seddiq Qais Abd Al-Rahman is a lecturer in Computer Science and Information Technology where he has been a college employee since 2007. He was Rapporteur of the Computer Network Systems Department. From 2007–2016, He taught many laboratory materials and programming languages. Abd Al-Rahman completed his M.Sc. degree at the University of Anbar at 2019 and his undergraduate studies at the same University. He teaches computer science in general, cryptographic, visual programing and structure programing. His areas of expertise interests are focused on the approaches for securing transfer data and store it with intelligent methods. He is also interested in handling with the smartphone applications. He can be contacted at email: co.sedeikaldossary@uoanbar.edu.iq.

Shihab A. Shawkat received the B.Sc. degree in Computer Science from University of Tikrit in 2007 and M.Sc. Degree in Computer Science from Mansoura University in 2017. Mr. Shihab A. Shawkat worked as a teacher during the period from 2008 to 2019 in Directorate of Education in Salah Al-Din, Ministry of Education, Iraq. He has recently started working at the University of Samarra at the end of 2019 till now. His research interest lies in computer science, information security, image processing and AI. He can be contacted at email: shahab84ahmed@gmail.com.