Linear Cryptoanalysis of the Simplified AES Cipher Modified by Chaotic Sequences

José A. P. Artiles, Daniel P. B. Chaves, Cecilio Pimentel

***Abstract*—This article introduces new symmetric key architectures based on a randomized version of the Simplified Advanced Encryption Standard (SAES). It is proposed a new technique to randomize the S-boxes of the original SAES employing chaotic sequences. Then, we study the linear criptanalysis of the proposed schemes. It is shown that, with the introduction of chaotic sequences, the adversary needs a larger number of pairs of plaintext and ciphertext to discover the bits of the key compared to the required by the SAES. Given these results, it is possible to evaluate the improvement of the proposed technique against linear cryptanalysis as compared to the original AES algorithm.**

***Index Terms*—Block ciphers, chaotic sequences, linear cryptanalysis, security analysis, simplified advanced encryption standard.**

I. Introdution

The Advanced Encryption Standard (AES) is the standard algorithm adopted by the National Institute of Standards and Technology (NIST) as its current recommendation for the symmetric key encryption algorithm [1]. The input block has 128 bits and the number of rounds varies depending on the key size, that is, an AES cipher with 128, 192 or 256-bit key works with 10, 12, or 14 rounds, respectively [2]. The AES has four units per round: SubBytes, ShiftRows, MixColumns, AddRoundKey and allows an efficient software implementation [3]–[5]. An important step of this algorithm is the SubBytes unit since it provides confusion in the ciphertext and is carried out by the S-boxes.

In general, the S-boxes are not sufficiently secure against cryptanalysis due to their rigid architecture [2]. This means that identical plaintext blocks are encrypted to identical ciphertext blocks when the same key is used. Therefore, techniques to improve the security of this unit have a prominent impact on the security of a block cipher. We propose in this work a randomized S-box employing chaotic sequences. These sequences are characterized by irregularity, aperiodicity, decorrelation, and broadband and can be generated through simple deterministic dynamical systems [6].

A set of security metrics (e.g. Shannon entropy, correlation coefficient, key sensitivity) [4], [7], [8] is commonly used to evaluate the randomness of the ciphertext and its capacity to resist statistical attacks. Other analyses should also be performed on cipher algorithms, such as their robustness against linear cryptanalysis (LC). This cryptanalysis is based on linear approximations of the nonlinear operations performed by the S-boxes. A precursor work in LC was introduced by Matsui [9] in 1992. In 1993, this technique was used as an attack on DES [10].

The computational effort to evaluate the effectiveness of the LC in the original AES algorithm can be prohibitive, as a solution, a simplified AES algorithm (SAES) was proposed in [11]. It has 2 rounds and the data input block is shorter than the original AES, without losing the essence of the original algorithm. This means that, by understanding the SAES algorithm and expanding its concepts, the behavior of this cryptanalysis in the AES algorithm can be understood. The objective of this work is to propose new block cipher architectures based on the SAES S-box modified by chaotic sequences and study the LC for these ciphers. The new schemes, namely SAES1, SAES2, SAES3, establish a compromise between computational complexity and security. It is shown that the new ciphers are considerably more robust against LC than the original SAES.

The rest of this article is organized in four sections. Section II describes the SAES algorithm. The LC for the SAES system is discussed in Section III and this analysis is extended to the algorithms SAES1, SAES2, SAES3 in Section IV. A comparison of the robustness of these systems against LC is made in this section. The conclusions of this work are summarized in Section V.

II. Preliminaries

A. Simplified AES algorithm

In the SAES [11], each input block (plaintext) has 16 bits $\{x_0, \cdots, x_{15}\}$ and the original key also has 16 bits $\{k_0, \cdots, k_{15}\}$. This cipher has two rounds, thus 2 subkeys are created from the original key, $\{k_{16}, \cdots, k_{31}\}$ and $\{k_{32}, \cdots, k_{47}\}$, totalizing 48 key bits.

In the first round, the original key is added (module 2) to the plaintext. The SAES has the same units as the original AES algorithm (SubBytes, ShiftRows, MixColumns, AddRoundKey). The second round has two units: SubBytes and AddRoundKey, as illustrated in Fig. 1. The output of SAES is the ciphertext $\{y_0, \cdots, y_{15}\}$. The operations performed in each round are described next.

1) **SubBytes**: The input bits to the SubBytes unit are given by $a_i = x_i \oplus k_i$, for $\{i = 0, \cdots, 15\}$, where $\oplus$ denotes addition modulo 2. This unit comprises 4 identical S-boxes operating in parallel, where each S-box has 4 input bits and 4 output bits.

Department of Electronics and Systems, Federal University of Pernambuco, Recife, PE, 50711-970, Brazil, email: {joseantonio.artiles,daniel.chaves,cecilio}@ufpe.br.

This work was supported in part by the Brazilian National Council for Scientific and Technological Development (CNPq) and by the Foundation for Support of Science and Technology of the State of Pernambuco (FACEPE).

A preliminary version of this paper was presented in XXXVI Simpósio Brasileiro de Telecomunicações e Processamento de Sinais (SBrT'18), Campina Grande, PB, Brazil, September 16-19, 2018.

Digital Object Identifier: 10.14209/jcis.2019.10

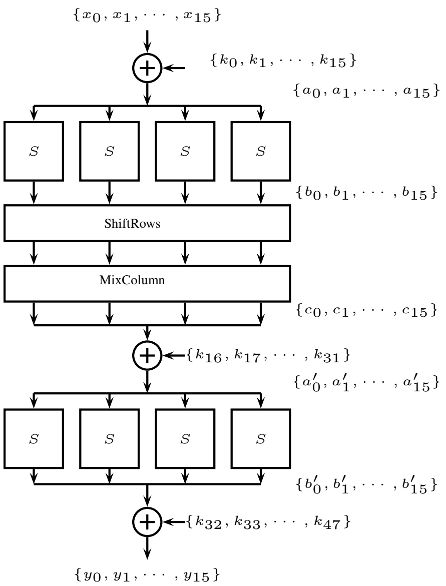
$\{x_0, x_1, \cdots, x_{15}\}$
$\{k_0, k_1, \cdots, k_{15}\}$
$\{a_0, a_1, \cdots, a_{15}\}$
S
S
S
S
$\{b_0, b_1, \cdots, b_{15}\}$
ShiftRows
MixColumn
$\{c_0, c_1, \cdots, c_{15}\}$
$\{k_{16}, k_{17}, \cdots, k_{31}\}$
$\{a'_0, a'_1, \cdots, a'_{15}\}$
S
S
S
S
$\{b'_0, b'_1, \cdots, b'_{15}\}$
$\{k_{32}, k_{33}, \cdots, k_{47}\}$
$\{y_0, y_1, \cdots, y_{15}\}$

Fig. 1. Block diagram of the SAES algorithm with two rounds.

TABLE I
INPUT-OUTPUT BITS OF AN S-BOX

Input	Output	Input	Output
$\{0,0,0,0\} \rightarrow$	$\{1,0,0,1\}$	$\{1,0,0,0\} \rightarrow$	$\{0,1,1,0\}$
$\{0,0,0,1\} \rightarrow$	$\{0,1,0,0\}$	$\{1,0,0,1\} \rightarrow$	$\{0,0,1,0\}$
$\{0,0,1,0\} \rightarrow$	$\{1,0,1,0\}$	$\{1,0,1,0\} \rightarrow$	$\{0,0,0,0\}$
$\{0,0,1,1\} \rightarrow$	$\{1,0,1,1\}$	$\{1,0,1,1\} \rightarrow$	$\{0,0,1,1\}$
$\{0,1,0,0\} \rightarrow$	$\{1,1,0,1\}$	$\{1,1,0,0\} \rightarrow$	$\{1,1,0,0\}$
$\{0,1,0,1\} \rightarrow$	$\{0,0,0,1\}$	$\{1,1,0,1\} \rightarrow$	$\{1,1,1,0\}$
$\{0,1,1,0\} \rightarrow$	$\{1,0,0,0\}$	$\{1,1,1,0\} \rightarrow$	$\{1,1,1,1\}$
$\{0,1,1,1\} \rightarrow$	$\{0,1,0,1\}$	$\{1,1,1,1\} \rightarrow$	$\{0,1,1,1\}$

The output bits are obtained through nonlinear and reversible operations defined in Galois field GF(2^4), generated by the primitive polynomial $P(x) = x^4 + x + 1$. Let a_0, a_1, a_2, a_3 be the input to an S-box. Initially, the multiplicative inverse of this sequence is determined in GF(2^4) (the sequence 0000 is not invertible, so the corresponding output is 0000). The inverted input sequence $a_0^-, a_1^-, a_2^-, a_3^-$ is used to obtain the output of the S-box (b_0, b_1, b_2, b_3) as

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} a_0^- \\ a_1^- \\ a_2^- \\ a_3^- \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}. \tag{1}$$

The mapping between the input and output bits of an S-box is shown in Table I.

2) **ShiftRows**: In this unit, the sequence $\{b_0, \cdots, b_3, b_4, \cdots, b_7, b_8, \cdots, b_{11}, b_{12}, \cdots, b_{15}\}$ is mapped into $\{b_0, \cdots, b_3, b_{12}, \cdots, b_{15}, b_8, \cdots, b_{11}, b_4, \cdots, b_7\}$.

3) **MixColumns**: The MixColumns unit performs a mixture of bits from the output of distinct S-boxes. This is the major diffusion step in the SAES. This assignment is given by [11]

$$\begin{array}{llll} c_0 = & b_0 \oplus b_{14} & c_8 = & b_6 \oplus b_8 \\ c_1 = & b_1 \oplus b_{12} \oplus b_{15} & c_9 = & b_4 \oplus b_7 \oplus b_9 \\ c_2 = & b_2 \oplus b_{12} \oplus b_{13} & c_{10} = & b_4 \oplus b_5 \oplus b_{10} \\ c_3 = & b_3 \oplus b_{13} & c_{11} = & b_5 \oplus b_{11} \\ c_4 = & b_2 \oplus b_{12} & c_{12} = & b_4 \oplus b_{10} \\ c_5 = & b_0 \oplus b_3 \oplus b_{13} & c_{13} = & b_5 \oplus b_8 \oplus b_{11} \\ c_6 = & b_0 \oplus b_1 \oplus b_{14} & c_{14} = & b_6 \oplus b_8 \oplus b_9 \\ c_7 = & b_1 \oplus b_{15} & c_{15} = & b_7 \oplus b_9. \end{array} \tag{2}$$

4) **AddRoundKey**: In this unit, the output bits of the MixColumns unit are added to the subkey bits $\{k_{16}, \cdots, k_{31}\}$, and this finalizes the first round. In the second round, the output bits of the SubBytes unit $\{b'_0, \cdots, b'_{15}\}$ are added to the subkeys bits $\{k_{32}, \cdots, k_{47}\}$ to generate the ciphertext $\{y_0, \cdots, y_{15}\}$, that is

$$y_i = b'_i + k_{i+32} \tag{3}$$

for $i = 0, \cdots, 15$.

5) **Subkeys Schedule**: Four S-boxes are used to obtain the two subkeys from the original key. These subkeys are given by

$$\begin{array}{rcl} k_{16} & = & k_0 \oplus l_0 \oplus 1 \\ k_i & = & k_{i-16} \oplus l_{i-16}, \quad i = 17, 18, \cdots, 23 \\ k_i & = & k_{i-16} \oplus l_{i-24}, \quad i = 32, 33, 36, \cdots, 39 \\ k_{34} & = & k_{18} \oplus l_{10} \oplus 1 \\ k_{35} & = & k_{19} \oplus l_{11} \oplus 1 \\ k_i & = & k_{i-8} \oplus k_{i-16}, \; i = 24, 25, \cdots, 31, 40, 41, \cdots, 47 \end{array} \tag{4}$$

where $(l_0, \cdots, l_{15})$ are the outputs of the 4 S-boxes, being related to the original key as

$$\begin{array}{rcl} S(k_{12}, k_{13}, k_{14}, k_{15}) & = & l_0 l_1 l_2 l_3 \\ S(k_8, k_9, k_{10}, k_{11}) & = & l_4 l_5 l_6 l_7 \\ S(k_{28}, k_{29}, k_{30}, k_{31}) & = & l_8 l_9 l_{10} l_{11} \\ S(k_{24}, k_{25}, k_{26}, k_{27}) & = & l_{12} l_{13} l_{14} l_{15}. \end{array} \tag{5}$$

It is worth observing that sixteen key bits determined by the last line in (4) are linear combinations of other key bits. Therefore, to obtain the 48 bits of the key it is only necessary to determine 32 of such bits.

B. Chaotic Maps

A binary sequence obtained from an one-dimensional chaotic maps is given by the iteration of a nonlinear and noninvertible function $f(x)$, under an initial condition x_0. Initially, a discrete-time series $\{x_i\}_{i=0}^{\infty}$ is generated according to [6]

$$x_n = f(x_{n-1}), \quad n = 1, 2, 3, \ldots, \tag{6}$$

generating an orbit $\{x_n\}_{n=0}^{\infty} = \{x_0, f(x_0), f(f(x_0)), \ldots\}$ of $f(x)$ starting at the initial condition x_0. Then, a binary sequence $\{z_n\}$, denoted by binary chaotic sequence, is obtained from $\{x_n\}$ via hard quantization [12].

Chaotic maps are known to generate uncorrelated, noise-like, aperiodic real valued sequences [6]. An important property of chaotic systems is that they are deeply sensitive on the initial condition of the system, meaning that nearby trajectories separate exponentially fast. A widely used metric to measure this sensitivity on initial conditions and determine whether the map evolves to a stable or chaotic behavior is the Lyapunov exponent. A chaotic systems has necessarily a positive Lyapunov [6].

In cryptography applications of chaotic systems, the value of x_0 is obtained from the original key. For a block cipher with key size of 128 bits, as the AES, the bits of this key are clustered into a block of 16 bytes, $v_1, v_2, \cdots, v_{16}$ (where v_i is the decimal representation of each byte) and let m_0' be defined as $m_0' = \sum_{i=1}^{16} \frac{v_i}{256}$. Then, $x_0 = m_0' - \lfloor m_0' \rfloor$, where $\lfloor \cdot \rfloor$ is the floor function. The first 200 samples of the orbit generated from x_0 are discarded to eliminate the transient behavior. Examples of chaotic maps $f : [-1, 1] \to [-1, 1]$ include the cubic map $f(x) = 4x^3 - 3x$ and the logistic map $f(x) = 4x(1 - x)$.

Due to the noise-like behavior of chaotic sequences it is hard to obtain useful information about the behavior of the sequences generated by a chaotic map from the observation of the time evolution. Despite being deterministic and defined by difference equations a chaotic map with uncertain initial condition can be characterized as a stochastic process, where the orbit of each initial condition under the map is a realization of the process.

III. Linear Cryptoanalysis

The LC explores linear relationships between the input and output bits of the S-boxes. Since the S-boxes are the nonlinear units of the SAES, the best that can be done is to find linear relations between input and output with distinguished probability. The LC is a known plaintext attack, that is, the adversary knows a set of pairs of plaintexts and the corresponding ciphertexts obtained with the same key. The idea of LC is to find linear equations of the form

$$\sum_{k \in S_1} x_k \oplus \sum_{l \in S_2} y_l = \left(\sum_{m \in S_3} k_m \right) \oplus t \tag{7}$$

with probability greater than 0.5, where t a bit with value 0 or 1, x_k is the k-th bit of plaintext, y_l is the l-th bit of ciphertext, k_m represents the m-th bit of the key and each S_i is a subset of $\{0, \cdots, 15\}$.

For each equation, the adversary evaluates the left-hand side of (7) for each plaintext-ciphertext pair and estimates the probability that the right-hand side is correct. Let p_ℓ be the probability that the ℓ-th equation is correct in such a way that the bit t is chosen so that $p_\ell \geq 0.5$. If a cipher shows a trend that (7) is satisfied with probability close to $1/2$, it is an evidence that it is robust for this cryptanalysis. The further away the probability p_ℓ is from $1/2$, the more effective is the LC.

TABLE II
Number of equations satisfied with probability p_ℓ

$p_\ell = 0.5$	$p_\ell = 0.5625$	$p_\ell = 0.6252$	$p_\ell = 0.75$
152	32	60	12

A. Linear Cryptoanalysis of the SAES

This section analyzes the LC for the SAES introduced in Subsection II-A. The main idea is to find linear equations corresponding to the input and output bits of the S-boxes that have probability greater than 0.5. Let us consider an S-box where the input and output bits are related as $S(a_0a_1a_2a_3) = b_0b_1b_2b_3$. There are 256 equations for all possible combinations of the input and output bits of this S-box and the following 12 equations occur with probability 0.75

$$\begin{array}{rlrl} a_0 \oplus b_2 &= 0 & a_1 \oplus b_0 \oplus b_3 &= 0 \\ a_1 \oplus b_1 &= 0 & a_2 \oplus a_3 \oplus b_3 &= 1 \\ a_0 \oplus a_1 \oplus b_1 \oplus b_2 \oplus b_3 &= 1 & a_3 \oplus b_0 &= 1 \\ a_0 \oplus a_1 \oplus a_2 \oplus a_3 \oplus b_1 &= 0 & a_2 \oplus b_2 \oplus b_3 &= 1 \\ a_2 \oplus b_1 \oplus b_3 &= 1 & a_1 \oplus a_2 \oplus b_0 \oplus b_1 &= 1 \\ a_0 \oplus a_1 \oplus b_0 &= 1 & a_0 \oplus b_0 \oplus b_1 &= 1. \end{array} \tag{8}$$

The number of equations for each possible probability is shown in Table II. Considering 4 S-boxes, 48 equations are obtained with probability 0.75 in the first round that depend on the plaintext bits, the key bits, and the output bits of this unit $\{b_0, \cdots, b_{15}\}$, as for example, $b_5 \oplus x_5 = k_5$ and $b_8 \oplus b_{11} \oplus x_9 = k_9$.

The adversary knows pairs of plaintext $\{x_0, \cdots, x_{15}\}$ and ciphertext $\{y_0, \cdots, y_{15}\}$, thus, it is necessary to perform the same procedure in the second SubBytes unit and combine the resulting equations with those obtained in first SubBytes unit through linear operations, that is, we write $\{a_1', \cdots, a_{15}'\}$ depending on $\{b_0, \cdots, b_{15}\}$, and in the same way $\{b_0', \cdots, b_{15}'\}$ as a function of $\{y_0, \cdots, y_{15}\}$, thus 48 equations are obtained in the second round, each having probability 0.75, such as $b_5 \oplus b_8 \oplus b_{11} \oplus y_{12} \oplus y_{15} = k_{29} \oplus k_{44} \oplus k_{47} \oplus 1$. To obtain equations of the form given in (7), a combination (sum module 2) of equations of each round must be performed, as for example

$$\begin{array}{c} b_5 \oplus x_5 = k_5 \\ b_8 \oplus b_{11} \oplus x_9 = k_9 \\ b_5 \oplus b_8 \oplus b_{11} \oplus y_{12} \oplus y_{15} = k_{29} \oplus k_{44} \oplus k_{47} \oplus 1 \\ \hline x_5 \oplus x_9 \oplus y_{12} \oplus y_{15} = k_5 \oplus k_9 \oplus k_{29} \oplus k_{44} \oplus k_{47} \oplus 1 \end{array}. \tag{9}$$

Since each equation in each round is satisfied with a certain probability, in the following subsection the probability of an equation obtained from the combination of other equations is calculated.

1) Combination of equations: The equations obtained from the S-boxes in the first and second rounds of the SubBytes units are considered binary random variables. Let X and Y be independent Bernoulli random variables associated with linear equations obtained from distinct S-boxes of the SAES (in the same round or in distinct rounds). The event $X = 1$ means that the equation is satisfied. The same holds for Y. Let $p_1 \triangleq$

$\Pr(X = 1)$ and $p_2 \triangleq \Pr(Y = 1)$, where $0.5 < p_1, p_2 < 1$. Now let V be a Bernoulli random variable such that $V = 1$ means that the linear combination of equations associated with X and Y is satisfied. Thus

$$\begin{aligned}\Pr(V = 1) &= \Pr(X = 1, Y = 1) + \Pr(X = 0, Y = 0)\\ &= \Pr(X = 1)\ \Pr(Y = 1) + \Pr(X = 0)\Pr(Y = 0)\\ &= p_1p_2 + (1 - p_1)(1 - p_2). \qquad (10)\end{aligned}$$

When $p_1 = p_2 = p$, we obtain

$$q \triangleq \Pr(V = 1) = 2p^2 - 2p + 1. \qquad (11)$$

For example, when $p = 0.75$, we get $q = 2(0.75)^2 - 2(0.75) + 1 = 0.625$. It is important to note that $0.5 \leq q \leq p$ when p is in the interval $0.5 \leq p \leq 1$, since

$$q - p = 2p^2 - 3p + 1 = (2p - 1)(p - 1). \qquad (12)$$

For the valid interval of p, the term $(p - 1)$ is negative while $(2p - 1)$ is positive, resulting that $q - p \leq 0$.

Following an analogous reasoning, it can be shown that the combination of equations with different probabilities is limited by

$$q \ = \ p_1p_2 + (1 - p_1)(1 - p_2) \leq \min(p_1, p_2). \qquad (13)$$

Therefore, the probability of the combination of equations with different probabilities is limited by the least of them, reaching the lowest value equal to 0.5 when one of the probabilities is 0.5. This method is adequate to determine an upper bound on the probability that an equation be satisfied, when it is generated from the combination of equations either from distinct S-boxes in the same round or from distinct rounds. Since, the plaintext bits and key bits that form these equations are independent binary random variables.

The combination of equations obtained in the first and second rounds of the SubBytes units results in equations of the form given in (7). Considering (9), the sum of $b_5 \oplus x_5 = k_5$ and $b_8 \oplus b_{11} \oplus x_9 = k_9$ (obtained in the first round) results in an equation with probability

$$\begin{aligned}q &= 2p^2 - 2p + 1\\ &= 2(0.75)^2 - 2(0.75) + 1 = 0.625 \qquad (14)\end{aligned}$$

that is added to $b_5 \oplus b_8 \oplus b_{11} \oplus y_{12} \oplus y_{15} = 1 \oplus k_{29} \oplus k_{44} \oplus k_{47}$ (second round) resulting in an equation with probability

$$\begin{aligned}q_1 &= p_1p_2 + (1 - p_1)(1 - p_2) \qquad (15)\\ &= (0.75)(0.625) + (1 - 0.75)(1 - 0.625)\\ &= 0.5625.\end{aligned}$$

Repeating this process for all the equations obtained in the two SubBytes units, we obtain the 32 linearly independent equations listed in the Appendix each one with probability 0.5625. An important question is how many pairs of plaintext-ciphertext n are necessary for the adversary to break the algorithm (with some reliability) using these 32 equations. We consider in this work a reliability of 95%.

Let W be a random variable that models the proportion of n pairs of plaintext-ciphertext for which the right hand side of each equation in the Appendix is the correct, for a certain key. Each pair of plaintext-ciphertext is a realization of an experiment with probability of correct equal to q. Each realization is independent and can be described by a binomial distribution normalized by n. So, the average value of W is q and its variance is

$$\sigma^2 = q(1 - q)/n. \qquad (16)$$

For the LC, it is desired that $\Pr(W \geq 0.5)$. Using the established reliability, we have that

$$\Pr(W \geq 0.5) = \sqrt[32]{0.95} = 0.9984.$$

For a sufficiently large n, W the Cumulative Distribution Function (CDF) of W tends to the CDF of a normal random variable. Defining a normal random variable $Z = (W - q)/\sigma$ with zero mean and unit variance, we have

$$\begin{aligned}\Pr(W \geq 0.5) &= \Pr\left(Z \geq \frac{0.5 - q}{\sigma}\right)\\ &= \Pr\left(Z \geq \frac{\sqrt{n}(0.5 - q)}{\sqrt{q(1 - q)}}\right)\\ &= \Pr\left(Z \geq -\frac{\sqrt{n}(q - 0.5)}{\sqrt{q(1 - q)}}\right) \qquad (17)\\ &= 1 - Q\left(\frac{\sqrt{n}(q - 0.5)}{\sqrt{q(1 - q)}}\right)\\ &= 0.9984\end{aligned}$$

where $Q(x) = 1/\sqrt{2\pi} \int_x^\infty \exp\{-t^2/2\}dt$ is the Gaussian Q-function. Therefore

$$Q\left(\frac{\sqrt{n}(q - 0.5)}{\sqrt{q(1 - q)}}\right) = 0.0016. \qquad (18)$$

For the case $q = 0.5625$, we obtain

$$Q(0.126\sqrt{n}) = 0.0016. \qquad (19)$$

The argument of the function $Q(x)$ that satisfies (19) is 2.94, then we obtain $n = 544.55$. In this way, 545 pairs of plaintext-ciphertext are needed to discover the bits of the key with a reliability of 95%. Thus, the LC is attractive compared to a pure brute force attack for the SAES with two rounds. In the next section, a similar analysis is performed for SAES algorithms modified by chaotic sequences.

IV. Linear Cryptoanalysis of the SAES Modified by a Chaotic Sequence

In this section, the complexity of the LC attack is analyzed for three proposed algorithms based on the SAES with the S-boxes modified by a chaotic sequence. These are called SAES1, SAES2, SAES3.

A. *SAES1*

In this algorithm, the 4 output bits of each S-box are added to a binary chaotic sequence $\mathbf{h}$ generated from a chaotic map. Two chaotic bits z_0 and z_1 are used in the S-boxes of the datapath and are represented in two equivalent forms; as a vector (z_0, z_1) or as a polynomial $c(x) = z_0x + z_1$. The polynomial $c(x)$ is multiplied by the primitive polynomial $p(x) = x^3 + x + 1$ in GF(2^4), obtaining a polynomial $h(x) = c(x)p(x) \mod P(x)$, where $P(x) = x^4 + x + 1$. The coefficients of this polynomial form a sequence $\mathbf{h} = (h_0, h_1, h_2, h_3)$. This mapping is given by

$$\begin{array}{rcl} (z_0, z_1) & \rightarrow & (h_0, h_1, h_2, h_3) \\ (0,0) & \rightarrow & (0,0,0,0) \\ (0,1) & \rightarrow & (1,0,1,1) \\ (1,0) & \rightarrow & (0,1,0,1) \\ (1,1) & \rightarrow & (1,1,1,0) \end{array} \quad (20)$$

Two chaotic bits (z_2, z_3) are used to obtain the bits of the subkeys, totalizing four chaotic bits to encrypt a plaintext of 16 bits. To simplify the analysis, the same chaotic sequence is used in the second round. It is observed from (20) that h_0 and h_2 are equal to z_1, h_1 is equal to z_0, and h_3 is equal to $z_0 \oplus z_1$. Therefore, the output bits of the S-boxes in the first round of the SAES1 are related to the output bits of the SAES algorithm as follows

$$[\hat{b}_i, \hat{b}_{i+1}, \hat{b}_{i+2}, \hat{b}_{i+3}] = [b_i, b_{i+1}, b_{i+2}, b_{i+3}] \oplus [z_1, z_0, z_1, z_0 \oplus z_1] \quad (21)$$

for $i \in \{0, 4, 8, 12\}$. In a similar way the output bits are obtained in the second round $(\hat{b}'_0, \cdots, \hat{b}'_{15})$.

In an anolog form, the subkeys $(\hat{k}_{16}, \cdots, \hat{k}_{47})$ of the SAES1 are related to the corresponding bits of the SAES as

$$[\hat{k}_i, \hat{k}_{i+1}, \hat{k}_{i+2}, \hat{k}_{i+3}] = [k_i, k_{i+1}, k_{i+2}, k_{i+3}] \oplus [z_3, z_2, z_3, z_2 \oplus z_3]$$

for $i \in \{16, 20, 24, 28, 32, 36, 40, 44\}$. The ciphertext is expressed as

$$[\hat{y}_i, \hat{y}_{i+1}, \hat{y}_{i+2}, \hat{y}_{i+3}] = [y_i, y_{i+1}, y_{i+2}, y_{i+3}] \oplus [z_1, z_0, z_1, z_0 \oplus z_1]$$

for $i \in \{0, 4, 8, 12\}$. The SAES1 algorithm has the same structure as the SAES, just replace b_i, y_i e k_i by $\hat{b}_i$, $\hat{y}_i$ and $\hat{k}_i$, respectively. For example, for two equations in the first round

$$\hat{b}_5 \oplus x_5 = k_5, \quad \text{or} \quad b_5 \oplus z_0 \oplus x_5 = k_5. \quad (22)$$

Analogously,

$$\hat{b}_8 \oplus \hat{b}_{11} \oplus x_9 = k_9, \quad \text{or} \quad b_8 \oplus z_1 \oplus b_{11} \oplus z_0 \oplus z_1 \oplus x_9 = k_9$$

or

$$b_8 \oplus b_{11} \oplus z_0 \oplus x_9 = k_9. \quad (23)$$

The equation of the second round $b_5 \oplus b_8 \oplus b_{11} \oplus y_{12} \oplus y_{15} = k_{29} \oplus k_{44} \oplus k_{47} \oplus 1$ of the algorithm SAES, becomes

$$\hat{b}_5 \oplus \hat{b}_8 \oplus \hat{b}_{11} \oplus \hat{y}_{12} \oplus \hat{y}_{15} = \hat{k}_{29} \oplus \hat{k}_{44} \oplus \hat{k}_{47} \oplus 1$$

or

$$\begin{aligned} b_5 \oplus z_0 \oplus b_8 \oplus z_1 \oplus b_{11} \oplus z_0 \oplus z_1 \oplus y_{12} \oplus z_1 \oplus y_{15} \oplus z_0 \oplus z_1 = \\ k_{29} \oplus z_2 \oplus k_{44} \oplus z_3 \oplus k_{47} \oplus z_2 \oplus z_3 \oplus 1 \end{aligned}$$

or

$$b_5 \oplus b_8 \oplus b_{11} \oplus y_{12} \oplus y_{15} = k_{29} \oplus k_{44} \oplus k_{47} \oplus z_0 \oplus 1. \quad (24)$$

Combining (22), (23), and (24), we get

$$\begin{array}{c} b_5 \oplus z_0 \oplus x_5 = k_5 \\ b_8 \oplus b_{11} \oplus z_0 \oplus x_9 = k_9 \\ b_5 \oplus b_8 \oplus b_{11} \oplus y_{12} \oplus y_{15} = k_{29} \oplus k_{44} \oplus k_{47} \oplus z_0 \oplus 1 \\ \hline x_5 \oplus x_9 \oplus y_{12} \oplus y_{15} = k_5 \oplus k_9 \oplus k_{29} \oplus k_{44} \oplus k_{47} \oplus z_0 \oplus 1 \end{array}. \quad (25)$$

In general, Equation (7) is modified to

$$\sum_{k \in S_1} x_k \oplus \sum_{l \in S_2} y_l = \left(\sum_{m \in S_3} k_m \right) \oplus \left(\sum_{r \in \Gamma} z_r \right) \oplus t \quad (26)$$

where Γ is a subset of $\{0, 1, 2, 3\}$. From the combinations of equations of each round, we obtain 48 equations that are divided into 9 groups, depending on the combination of chaotic bits in each equation. The number of equations in each group is shown in Table III. For example, the 8 equations of the group z_0 are

$$\begin{aligned} x_1 \oplus x_{13} \oplus y_0 \oplus y_3 &= k_1 \oplus k_{13} \oplus k_{17} \oplus k_{32} \oplus \\ &\quad k_{35} \oplus z_0 \\ x_1 \oplus x_{13} \oplus y_4 \oplus y_7 &= k_1 \oplus k_{13} \oplus k_{21} \oplus k_{36} \oplus \\ &\quad k_{39} \oplus z_0 \\ x_5 \oplus x_9 \oplus y_8 \oplus y_{11} &= k_5 \oplus k_9 \oplus k_{25} \oplus k_{40} \oplus \\ &\quad k_{43} \oplus z_0 \\ x_5 \oplus x_9 \oplus y_{12} \oplus y_{15} &= k_5 \oplus k_9 \oplus k_{29} \oplus k_{44} \oplus \\ &\quad k_{47} \oplus z_0 \\ x_1 \oplus x_{13} \oplus y_1 &= k_1 \oplus k_{13} \oplus k_{17} \oplus k_{33} \oplus z_0 \\ x_1 \oplus x_{13} \oplus y_5 &= k_1 \oplus k_{13} \oplus k_{21} \oplus k_{37} \oplus z_0 \\ x_5 \oplus x_9 \oplus y_9 &= k_5 \oplus k_9 \oplus k_{25} \oplus k_{41} \oplus z_0 \\ x_5 \oplus x_9 \oplus y_{13} &= k_5 \oplus k_9 \oplus k_{29} \oplus k_{45} \oplus z_0. \end{aligned}$$

The 4 equations of the group $z_1 \oplus z_2$ are

$$\begin{aligned} x_1 \oplus x_3 \oplus x_{12} \oplus x_{13} \oplus y_0 &= k_1 \oplus k_3 \oplus k_{12} \oplus k_{13} \oplus k_{16} \oplus \\ &\quad k_{17} \oplus k_{32} \oplus z_1 \oplus z_2 \\ x_0 \oplus x_1 \oplus x_{13} \oplus x_{15} \oplus y_4 &= k_0 \oplus k_1 \oplus k_{13} \oplus k_{15} \oplus k_{20} \oplus \\ &\quad k_{21} \oplus k_{36} \oplus z_1 \oplus z_2 \\ x_4 \oplus x_5 \oplus x_9 \oplus x_{11} \oplus y_8 &= k_4 \oplus k_5 \oplus k_9 \oplus k_{11} \oplus k_{24} \oplus \\ &\quad k_{25} \oplus k_{40} \oplus z_1 \oplus z_2 \\ x_5 \oplus x_7 \oplus x_8 \oplus x_9 \oplus y_{12} &= k_5 \oplus k_7 \oplus k_8 \oplus k_9 \oplus k_{28} \oplus \\ &\quad k_{29} \oplus k_{44} \oplus z_1 \oplus z_2 \end{aligned}$$

and the equations of the group $z_0 \oplus z_1 \oplus z_2$ are

$$\begin{aligned} x_3 \oplus x_{12} \oplus y_0 \oplus y_1 &= k_3 \oplus k_{12} \oplus k_{16} \oplus k_{32} \oplus k_{33} \oplus \\ &\quad z_0 \oplus z_1 \oplus z_2 \\ x_0 \oplus x_{15} \oplus y_4 \oplus y_5 &= k_0 \oplus k_{15} \oplus k_{20} \oplus k_{36} \oplus k_{37} \oplus \\ &\quad z_0 \oplus z_1 \oplus z_2 \\ x_4 \oplus x_{11} \oplus y_8 \oplus y_9 &= k_4 \oplus k_{11} \oplus k_{24} \oplus k_{40} \oplus k_{41} \oplus \\ &\quad z_0 \oplus z_1 \oplus z_2 \\ x_7 \oplus x_8 \oplus y_{12} \oplus y_{13} &= k_7 \oplus k_8 \oplus k_{28} \oplus k_{44} \oplus k_{45} \oplus \\ &\quad z_0 \oplus z_1 \oplus z_2. \end{aligned}$$

TABLE III
GROUPS OF EQUATIONS DEPENDING ON THE CHAOTIC BITS FOR SAES1

Chaotic Bits	Number of equations
z_0	8
z_1	8
z_2	4
z_3	4
$z_0 \oplus z_1$	4
$z_1 \oplus z_2$	4
$z_2 \oplus z_3$	8
$z_0 \oplus z_1 \oplus z_2$	4
$z_0 \oplus z_2 \oplus z_3$	4

Considering that the chaotic bits are independent and identically distributed random variables, the probability that an equation of the form (26) is satisfied is 0.5, thus the linear cryptanalysis cannot be applied in this case. However there are combinations of equations of distinct groups (listed in Table III) that allow us to obtain 32 linearly independent equations without chaotic bits. In the sequel, we show the required combinations and calculate the corresponding probabilities of the resulting equations without considering the chaotic bits (this calculation is performed in the same way as in the SAES algorithm), since the objective is to calculate (after all combinations) the probability of equations that do not involve chaotic bits. For example, from the SAES algorithm, each equation in Table III is satisfied with probability 0.5625. Thus, the addition modulo 2 of equations of the groups z_0 and $z_1 \oplus z_2$, yields 32 equations with probability

$$\begin{aligned} q_1 &\triangleq 2q^2 - 2q + 1 \\ &= 2(0.5625)^2 - 2(0.5625) + 1 = 0.5078 \end{aligned}$$

and adding these equations with those of the group $z_0 \oplus z_1 \oplus z_2$, we obtain a sufficient number of linearly independent equations that do not depend on the chaotic bits, each one with probability

$$\begin{aligned} q_2 &= (0.5625)(0.5078) + (1 - 0.5625)(1 - 0.5078) \\ &= 0.5009. \end{aligned} \quad (27)$$

Using this probability, we found that the adversary needs $n = 2,667,777$ pairs of plaintext-ciphertext (to find this value of n, we substitute $q = 0.5009$ into (17) and proceed in a similar way as in the paragraph after (19)). Another combinations of equations can be obtained, but the resulting probabilities are closer to 0.5, which increase the value of n. In summary, the chaotic bits select the groups of equations to be combined, while the probabilities of the equations after the combinations are calculated in the same way as in the SAES. We apply next this methodology for a SEAS2.

B. SAES2

The SAES2 algorithm is a simplified version of SAES1 in which the MixColumns unit is eliminated. Due to this elimination, the equations obtained in the second round of the SubBytes unit are modified. For example, one equation of the second round with probability 0.75 is $\hat{b}_5 \oplus \hat{y}_{12} \oplus \hat{y}_{15} = \hat{k}_{29} \oplus \hat{k}_{44} \oplus \hat{k}_{47}$. The combination of two equations of distinct rounds (each one with probability 0.75) results in 48 equations divided in 4 groups, as shown in the Table IV, each one with probability 0.625 (this probability is calculated in (14)). The combination of equations of the groups $z_0 \oplus z_2$ and $z_1 \oplus z_3$ leads to 96 equations of the group $z_0 \oplus z_1 \oplus z_2 \oplus z_3$ with probability 0.53125, which are combined with equations of the group $z_0 \oplus z_1 \oplus z_2 \oplus z_3$, resulting in equations with probability 0.5078 that do not depend on the chaotic bits. Using this probability, we find that the adversary needs $n = 35,518$ pairs of plaintext-ciphertext to find the key with reliability 95 %.

TABLE IV
GROUPS OF EQUATIONS DEPENDING ON THE CHAOTIC BITS FOR SAES2

Chaotic Bits	Number of equations
No	20
$z_0 \oplus z_2$	12
$z_1 \oplus z_3$	8
$z_0 \oplus z_1 \oplus z_2 \oplus z_3$	8

C. SAES3

The removal of the MixColumns unit of the SAES2 algorithm leads to a loss of diffusion of bits from distinct S-boxes. A new algorithm, namely SAES3, aims to compensate this effect. In this algorithm, the ShiftRows and MixColumns units are replaced by a new unit called ShiftRandom. A random cyclic shift to the right by j bits is performed on the output bits of the 4 S-boxes $(b_0, \cdots, b_{15})$ depending on the base ten value of the two chaotic bits $z_0 z_1$, for $j = 0, 1, 2, 3$. Each shift occurs with the same probability $1/4$ and for each one there are 48 possible equations. Table V shows the number of equations that depend on the chaotic bits for each shift, where these equations have probability either 0.625 or 0.5625.

A set of 32 linearly independent equations can be obtained from the combination of equations that are affected by chaotic bits. For example, a combination of the groups z_0 and z_1 in Table V, with shift 01 results in 16 equations that depend on $z_1 \oplus z_0$ with probability 0.5078. Combining these 16 equations with 8 equations that depend on $z_0 \oplus z_1$ (for the same shift) resulting in 128 equations, being possible to extract 32 linearly independent equations with probability 0.5009. This procedure can also be performed for the shifts 10 and 11 with the determination of 32 linearly independent equations with probability 0.5009 which do not depend on the chaotic bits. The procedures performed with the shift 00 are similar to those in the SAES2 algorithm, and the 32 equations present the probability 0.5078. Thus, the mean value of the probability of obtaining 32 linearly independent equations that do not depend on the chaotic bits is

$$q = \frac{1}{4}(0.5078) + \frac{3}{4}(0.5009) = 0.5026. \quad (28)$$

Following the derivation in Section III, the adversary needs $n = 319,660$ pairs of plaintext-ciphertext. Table VI summarizes the results of this section. It presents a comparison of the probabilities of linearly independent equations and the number of pairs of plaintext-ciphertext required for the LC to be successful with a reliability of 95%.

TABLE V
GROUPS OF EQUATIONS DEPENDING ON THE CHAOTIC BITS FOR SAES3

Shift	Chaotic bits	Number of equations	Probability
00	No	20	0.625
	$z_0 \oplus z_2$	12	0.625
	$z_1 \oplus z_3$	8	0.625
	$z_0 \oplus z_1 \oplus z_2 \oplus z_3$	8	0.625
01	z_0	4	0.5625
	z_1	4	0.5625
	z_2	4	0.5625
	$z_0 \oplus z_1$	8	0.5625
	$z_1 \oplus z_2$	4	0.5625
	$z_1 \oplus z_3$	4	0.625
	$z_2 \oplus z_3$	4	0.5625
	$z_1 \oplus z_2 \oplus z_3$	4	0.625
	$z_0 \oplus z_2 \oplus z_3$	4	0.625
	$z_0 \oplus z_1 \oplus z_3$	8	0.5625
10	z_1	8	0.625
	z_3	8	0.625
	$z_0 \oplus z_2$	4	0.5625
	$z_0 \oplus z_3$	8	0.5625
	$z_1 \oplus z_2$	8	0.5625
	$z_0 \oplus z_1 \oplus z_2$	8	0.625
	$z_0 \oplus z_1 \oplus z_2 \oplus z_3$	4	0.5625
11	z_0	8	0.5625
	z_2	8	0.5625
	$z_0 \oplus z_1$	4	0.5625
	$z_2 \oplus z_3$	4	0.5625
	$z_0 \oplus z_3$	4	0.5625
	$z_0 \oplus z_1 \oplus z_3$	8	0.5625
	$z_0 \oplus z_2 \oplus z_3$	4	0.5625
	$z_1 \oplus z_2 \oplus z_3$	8	0.625

TABLE VI
LC RESULTS FOR THE PROPOSED ALGORITHMS

Algorithm	probability	pairs
SAES	0.5625	545
SAES1	0.5009	2,667,777
SEAS2	0.5078	35,518
SAES3	0.5026	319,660

The introduction of chaotic bits leads to a considerable increase in the amount of pairs of plaintext-ciphertext compared to the required by the SAES algorithm. The SAES1 algorithm presents the best performance, but it is the most complex algorithm. The SAES3 algorithm presents robustness against LC significantly better than SAES2 with similar complexity (the only difference between these is the shift in the SAES3 that depends on the chaotic sequence).

V. CONCLUSIONS

We study the LC for modified SAES algorithms with the introduction of chaotic bits in the SubBytes and the generation of subkeys units. The new algorithms increase the number of pairs of plaintext-ciphertext needed to find the key bits with some reliability. As a future work, a similar analysis can conducted for other cryptanalysis techniques, such as differential cryptanalysis [13]. Another interesting future direction is to study the application of the proposed algorithms to some wireless protocols [14], [15].

APPENDIX

A set of 32 linearly independent equations of the SAES algorithm each one with probability 0.5625.

$$
\begin{aligned}
x_3 \oplus x_{12} \oplus y_2 &= k_3 \oplus k_{12} \oplus k_{16} \oplus k_{34} \oplus 1 \\
x_0 \oplus x_{15} \oplus y_6 &= k_0 \oplus k_{15} \oplus k_{20} \oplus k_{38} \oplus 1 \\
x_4 \oplus x_{11} \oplus y_{10} &= k_4 \oplus k_{11} \oplus k_{24} \oplus k_{42} \oplus 1 \\
x_7 \oplus x_8 \oplus y_{14} &= k_7 \oplus k_8 \oplus k_{28} \oplus k_{46} \oplus 1 \\
x_2 \oplus x_{15} \oplus y_3 &= k_2 \oplus k_{15} \oplus k_{18} \oplus k_{19} \oplus k_{35} \oplus 1 \\
x_3 \oplus x_{14} \oplus y_7 &= k_3 \oplus k_{14} \oplus k_{22} \oplus k_{23} \oplus k_{39} \oplus 1 \\
x_7 \oplus x_{10} \oplus y_{11} &= k_7 \oplus k_{10} \oplus k_{26} \oplus k_{27} \oplus k_{43} \oplus 1 \\
x_6 \oplus x_{11} \oplus y_{15} &= k_6 \oplus k_{11} \oplus k_{30} \oplus k_{31} \oplus k_{47} \oplus 1 \\
x_0 \oplus x_{12} \oplus y_2 \oplus y_3 &= k_0 \oplus k_{12} \oplus k_{18} \oplus k_{34} \oplus k_{35} \\
x_0 \oplus x_{12} \oplus y_6 \oplus y_7 &= k_0 \oplus k_{12} \oplus k_{22} \oplus k_{38} \oplus k_{39} \\
x_4 \oplus x_8 \oplus y_{10} \oplus y_{11} &= k_4 \oplus k_8 \oplus k_{26} \oplus k_{42} \oplus k_{43} \\
x_4 \oplus x_8 \oplus y_{14} \oplus y_{15} &= k_4 \oplus k_8 \oplus k_{30} \oplus k_{46} \oplus k_{47} \\
x_1 \oplus x_{13} \oplus y_0 \oplus y_3 &= k_1 \oplus k_{13} \oplus k_{17} \oplus k_{32} \oplus \\
&\quad k_{35} \oplus 1 \\
x_1 \oplus x_{13} \oplus y_4 \oplus y_7 &= k_1 \oplus k_{13} \oplus k_{21} \oplus k_{36} \oplus \\
&\quad k_{39} \oplus 1 \\
x_5 \oplus x_9 \oplus y_8 \oplus y_{11} &= k_5 \oplus k_9 \oplus k_{25} \oplus k_{40} \oplus \\
&\quad k_{43} \oplus 1 \\
x_5 \oplus x_9 \oplus y_{12} \oplus y_{15} &= k_5 \oplus k_9 \oplus k_{29} \oplus k_{44} \oplus \\
&\quad k_{47} \oplus 1 \\
x_2 \oplus x_3 \oplus x_{13} \oplus y_0 &= k_2 \oplus k_3 \oplus k_{13} \oplus k_{19} \oplus k_{32} \\
x_1 \oplus x_{14} \oplus x_{15} \oplus y_4 &= k_1 \oplus k_{14} \oplus k_{15} \oplus k_{23} \oplus k_{36} \\
x_5 \oplus x_{10} \oplus x_{11} \oplus y_8 &= k_5 \oplus k_{10} \oplus k_{11} \oplus k_{27} \oplus k_{40} \\
x_6 \oplus x_7 \oplus x_9 \oplus y_{12} &= k_6 \oplus k_7 \oplus k_9 \oplus k_{31} \oplus k_{44} \\
x_1 \oplus x_{13} \oplus y_1 &= k_1 \oplus k_{13} \oplus k_{17} \oplus k_{33} \\
x_1 \oplus x_{13} \oplus y_5 &= k_1 \oplus k_{13} \oplus k_{21} \oplus k_{37} \\
x_5 \oplus x_9 \oplus y_9 &= k_5 \oplus k_9 \oplus k_{25} \oplus k_{41} \\
x_5 \oplus x_9 \oplus y_{13} &= k_5 \oplus k_9 \oplus k_{29} \oplus k_{45} \\
x_0 \oplus x_1 \oplus x_{14} \oplus y_0 \oplus y_1 &= k_0 \oplus k_1 \oplus k_{14} \oplus k_{17} \oplus k_{18} \oplus \\
&\quad k_{32} \oplus k_{33} \oplus 1 \\
x_2 \oplus x_{12} \oplus x_{13} \oplus y_4 \oplus y_5 &= k_2 \oplus k_{12} \oplus k_{13} \oplus k_{21} \oplus k_{22} \oplus \\
&\quad k_{36} \oplus k_{35} \oplus 1 \\
x_6 \oplus x_8 \oplus x_9 \oplus y_8 \oplus y_9 &= k_6 \oplus k_8 \oplus k_9 \oplus k_{25} \oplus k_{26} \oplus \\
&\quad k_{40} \oplus k_{41} \oplus 1 \\
x_4 \oplus x_5 \oplus x_{10} \oplus y_{12} \oplus y_{13} &= k_4 \oplus k_5 \oplus k_{10} \oplus k_{29} \oplus k_{30} \oplus \\
&\quad k_{44} \oplus k_{45} \oplus 1 \\
x_3 \oplus x_{12} \oplus y_0 \oplus y_1 &= k_3 \oplus k_{12} \oplus k_{16} \oplus k_{32} \oplus k_{33} \\
x_0 \oplus x_{15} \oplus y_4 \oplus y_5 &= k_0 \oplus k_{15} \oplus k_{20} \oplus k_{36} \oplus k_{37} \\
x_4 \oplus x_{11} \oplus y_8 \oplus y_9 &= k_4 \oplus k_{11} \oplus k_{24} \oplus k_{40} \oplus k_{41} \\
x_7 \oplus x_8 \oplus y_{12} \oplus y_{13} &= k_7 \oplus k_8 \oplus k_{28} \oplus k_{44} \oplus k_{45}.
\end{aligned}
$$

REFERENCES

[1] F. I. P. S. (FIPS), "Advanced encryption standard (AES)," NIST, Tech. Rep. FIPS 197, Nov. 2001.

[2] C. Paar and J. Pelzl, Understanding Cryptography, A Textbook for Students and Practitioners. Springer, 2010.

[3] H. Li, “Efficient and flexible architecture for AES,” *IEE Proceedings - Circuits, Devices and Systems*, vol. 153, no. 6, pp. 533-538, Dec. 2006, doi: 10.1049/ip-cds:20050296.

[4] M. Preishuber, T. Htter, S. Katzenbeisser, and A. Uhl, “Depreciating motivation and empirical security analysis of chaos-based image and video encryption,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 9, pp. 2137-2150, Mar. 2018, doi: 10.1109/TIFS.2018.2812080.

[5] Y. Zhang, X. Li, and W. Hou, “A fast image encryption scheme based on AES,” in Proc. 2nd International Conference on Image, Vision and Computing (ICIVC), Chengdu, China, 2017, pp. 2624-2628, doi: 10.1109/ICIVC.2017.7984631.

[6] S. Strogatz, Nonlinear Dynamics and Chaos with Applications to Physics, Biology, Chemistry, and Engineering, ser. Studies in Nonlinearity Series. Westview Press, 2001.

[7] X. Wang and C. Liu, “A novel and effective image encryption algorithm based on chaos and DNA encoding,” *Multimedia Tools and Applications*, vol. 76, no. 5, pp. 6229-6245, Mar. 2017, doi: https://doi.org/10.1007/s11042-016-3311-8.

[8] J. Guo, D. Riyono, and H. Prasetyo, “Improved beta chaotic image encryption for multiple secret sharing,” *IEEE Access*, vol. 6, pp. 46297-46321, July 2018, doi: 10.1109/ACCESS.2018.2863021.

[9] M. Matsui and A. Yamagishi, “A new method for known plaintext attack of FEAL cipher,” in Proc. Workshop on the Theory and Application of of Cryptographic Techniques, 1992, pp. 81-91, doi: https://doi.org/10.1007/3-540-47555-9-7.

[10] M. Matsui, Linear cryptanalysis method for DES cipher, in Proc. Advances in Cryptology (EUROCRYPT 93), vol. 765, 1993, pp. 386-397, doi: https://doi.org/10.1007/3-540-48285-7-33.

[11] M. Musa, E. Schaefer, and S. Wedig, “A simplified AES algorithm and its linear and differential cryptanalysis,” *Cryptologia*, vol. 27, no. 2, pp. 148-177, 2003, doi: https://doi.org/10.1080/0161-110391891838.

[12] J. V. Evangelista, J. A. Artiles, D. P. Chaves, and C. Pimentel, “Emitter-coupled pair chaotic generator circuit,” *AEU - International Journal of Electronics and Communications*, vol. 77, pp. 112-117, 2017, doi: https://doi.org/10.1016/j.aeue.2017.04.029.

[13] J. Kim, S. Hong, and J. Lim, “Impossible differential cryptanalysis using matrix method”, *Discrete Mathematics*, vol. 310, no. 5, pp. 988-1002, Mar. 2010, doi: https://doi.org/10.1016/j.disc.2009.10.019.

[14] Y. L. Huang, F. Y. Leu, P. H. Su, T. H. Sung, and S. C. Liu, “A secure and high performance wireless sensor network based on symmetric key matrix,” in Proc. Int. Conf. Innov. Mobile Internet Services Ubiquitous Comput., Jul. 2016, pp. 470-475, doi: 10.1109/IMIS.2016.127.

[15] K. L. Tsai, Y. L. Huang, F. Y. Leu, I. You, Y. L. Huang, and C. H. Tsai, “AES-128 based secure low power communication for LoRaWAN IoT environments,” *IEEE Access*, vol. 6, pp. 45325-45334, 2018, doi: 10.1109/ACCESS.2018.2852563.

José Artiles received the B.Sc. degree in electrical engineering from the Central University of Las Villas, Villa Clara, Cuba, in 2008 and the M.Sc. degree in electrical engineering from the Federal University of Pernambuco, Recife, Brazil, in 2016. He is currently a doctoral student at the Federal University of Pernambuco. His current research interests include chaos communication, cryptography, watermarking, and random number generators.

Daniel Chaves received the the B.S. degree in electronics engineering and the M.S. degree in electrical engineering from the Federal University of Pernambuco, Recife, Brazil, in 2004 and 2006, respectively, and the Ph.D. degree in electrical engineering from the State University of Campinas, So Paulo, Brazil, in 2011. In 2012, he joined the Department of Electronics and Systems, Federal University of Pernambuco, Recife, Brazil, as an Assistant Professor. His current interests include information theory, coding theory, symbolic dynamics, system modeling, chaos communication, chaotic circuits and chaos based random number generators.

Cecilio Pimentel was born in Recife, Brazil, in 1966. He received the B.Sc. degree from the Federal University of Pernambuco, Recife, Brazil, in 1987; the M.Sc. degree from the Catholics University of Rio de Janeiro, Rio de Janeiro, Brazil, in 1990; and the Ph.D. degree from the University of Waterloo, Ontario, Canada, in 1996, all in electrical engineering. Since October 1996, he has been with the Department of Electronics and Systems at the Federal University of Pernambuco, where he is currently a Full Professor. From 2007 to 2008, he was a Visiting Research Scholar at the Department of Mathematics and Statistics, Queen’s University, Kingston, Canada. He is an IEEE Senior Member and a Senior Member of the Brazilian Telecommunications Society. His research interests include digital communications, information theory, chaos communication, and error correcting coding.