A Lightweight Cipher with Integrated Authentication

Eduardo Marsola do Nascimento[1], José Antônio Moreira Xexéo[2]

[1] Petróleo Brasileiro S.A. – Petrobras
Rio de Janeiro, RJ

[2]Instituto Militar de Engenharia
Rio de Janeiro, RJ

edunasci@yahoo.com, xexeo@ime.eb.br

Abstract. *This paper describes a symmetrical block cipher tailored to be used on Internet of Things (IoT) environment. It was engineered to be lightweight, consuming less computational resources than other ciphers, like AES, and to work with different block and key sizes. Other important characteristic is to integrate the authentication on its basic algorithm. This approach is helps to reduce the resource needs. The algorithm capacity to resist against linear and different cryptanalysis attacks and to generate was verified. The algorithm was also compared to 23 other ciphers implementations using the metrics generated by the FELICS (DINU et al., 2015) framework. The cipher randomness was also analyzed, using statistical tests.*

1. Introduction

On the IoT environment, where there is a plethora of devices, some hardware may be constrained in relation to program size, RAM requirements, processing power, execution time, chip area, energy consumption etc. For these devices, while, on one hand, there is a logical effort to minimize the resource use, on the other hand, the use of cryptography on them is a must. This observation opens our eyes to research lightweight cryptography, which is defined by ISO (ISO/IEC 29192-1:2012) as *"cryptography tailored for implementation in constrained environments"*.

This article describes the main result of our research, which is a new lightweight symmetrical cipher that was called FlexAE. It incorporates authentication on its most basic mode of operation. It was considered flexible because it works with diverse configurations using variable size for blocks (64×2^x bits) and keys (128×2^x bits, where $x \geq 0$).

Using differential cryptanalysis, the difficulty of an attack was calculated. It showed that the number of chosen plaintext necessary for the attack were bigger than the number of guesses for a successful brute force attack, making this kind of attack impractical. Using linear cryptanalysis, the number of chosen plaintext blocks for an attack was bigger than the possible number of blocks, when working with 64 and 128 block size, making the attack impossible. When using 256 and 512 block sizes the attack is possible. But the number chosen plaintext blocks necessary for the attack are 2^{254} and 2^{290}, making this kind of attack also impractical.

The FELICS (Fair Evaluation of Lightweight Cryptographic Systems) framework was used to compare the algorithm to other ciphers. This framework uses code size in bytes, RAM use in bytes and execution time in processor cycles as metrics.

The comparison showed that FlexAE has similar results as other lightweight ciphers.

The randomness of the cipher was also tested using statistical tests. It was applied the NIST statistical test suite following the process used to validate AES candidate selection by NIST (SOTO, 1999). The cipher was also tested using the statistical tests from dieharder tool (BROWN, 2016). The results confirm the randomness of the cipher.

1.1. Motivation and Objective

The motivation for this work was the observation that the IoT environment has very constrained devices that may require special lighter algorithms to work properly.

The main objective for this work was to propose a new cipher specially tailored to be used on IoT environment.

1.2. Contributions and Publications

The work described on this article is the main part of my master thesis: "Algoritmo de Criptografia Leve com Utilização de Autenticação" available online at http://www.comp.ime.eb.br/pos/arquivos/publicacoes/dissertacoes/2017/2017-Eduardo.pdf (NASCIMENTO, 2017).

I was advised by professor Dr. José Antonio Moreira Xexéo from Instituto Militar de Engenharia. It was presented and approved on April 28th 2017 in Rio de Janeiro. The committee was composed by Dr. Anderson Fernandes Pereira dos Santos, Dr. Flávio Luis de Mello and Dr. William Augusto Rodrigues de Souza.

Part of this work was publish on the article "A flexible authenticated lightweight cipher using Even-Mansour construction" presented at IEEE International Conference on Communications (ICC), Paris, 2017 (NASCIMENTO and XEXEO, 2017). This conference is classified as Qualis A1 by CAPES (Comissão de Aperfeiçoamento de Pessoal do Nível Superior).

1.3. Related Works

There are several works proposing lightweight ciphers. During the FlexAE development, various ciphers were surveyed like SKINNY (BEIERLE et al., 2016), MANTIS (BEIERLE et al., 2016), SIMON (BEAULIEU et al., 2015), PRINCE (BORGHOFF et al., 2012) and LED (GUO et al., 2011).Although it is very difficult endeavor to keep, an updated list of lightweight ciphers maintained by the CryptoLUX research group site from University of Luxembourg (CRYPTOLUX, 2016).

One original characteristic of FlexAE cipher that keeps it apart from other lightweight algorithm is the fact that it integrates the authentication. This construction was based on the Integrity Aware Parallelizable Mode (IAPM) (JUTLA, 2001).

2. Algorithm Description

The FleaxAE algorithm uses as a main component a key dependable permutation function(PF_K). On this function, like in the Even-Mansour cipher (EVEN and MANSOUR, 1997), the input is XORed with a key K_A at the beginning and with a key K_B at the end of the process.

Then it goes through a block shuffle layer, where a 2^{nb} bytes input is divided in $4\ bits$ blocks $(b[0], b[1], \dots, b[2^{nb+1}-1])$ and reordered as $(\mathrm{b}[0], \mathrm{b}[2^{\mathrm{nb}}], \mathrm{b}[1], \mathrm{b}[2^{\mathrm{nb}}+1], \dots, \mathrm{b}[2^{\mathrm{nb}}-1],\ \mathrm{b}[2^{\mathrm{nb}+1}-1]$.

The output goes into construction that resembles a Feistel network. This uses a SBox Layer to perform a non-linear transformation. Each byte of the input is submitted to one SBox. There are three different SBoxes. So the first byte is transformed by SBox0, the second by SBox1, the third by SBox2, the fourth by SBox0, and so on.

The SBox0 is the same used in AES Algorithm. As defined on the AES cipher specification (DAEMEN and RIJMEN, 2001), to create this SBox0, first a table is generated using the multiplicative inverse on the $Galois\ Field(2^8)$ defined by a specific irreducible polynomial ($p = x^8 + x^4 + x^3 + x^1 + 1$). The table then suffers an affine transformation. It is multiplied by $0x1F\ mod\ (x^8 + 1)$ and it is added the constant $0x63$. The other two other are created using the same process but using different parameters.

Table 1. SBoxes parameters

# SBox	Irreducible Polynomial	Multiplicative Constant	Aditive Constant
0	$x^8 + x^4 + x^3 + x^1 + 1$	0x1F	0x63
1	$x^8 + x^4 + x^3 + x^2 + 1$	0x1F	0x95
2	$x^8 + x^5 + x^3 + x^1 + 1$	0x1F	0xA6

The number of rounds (r) that the data goes through the Feistel like construction is $r = \log_2 nb + 2$, where nb=block size in bytes.

The FlexAE cipher uses three subkeys (K_1, K_2, K_3). They are created from a bit sequence generates by applying the permutation function twice using the main key K (PF_K) until have enough bits for all subkeys. The initial value is a sequence of zeros $(0^{ks/2})$. Each subkey (K_1, K_2, K_3) size is $2 \times \mathrm{nb}$, which is double the block size in bytes (or $16 \times nb$ in bits). The main key K size is $128 \times 2^{\mathrm{x}}$ bits , where $\mathrm{x} \geq 0$.

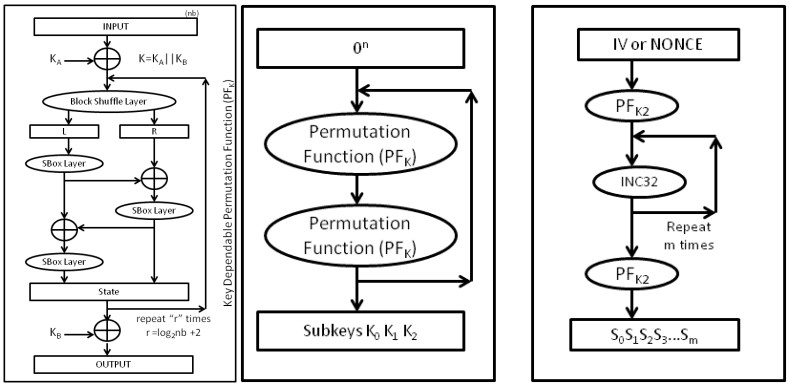

Figure 1. The permutation function (PF), the K_0,K_1,K_2 and $S_0S_1 \dots S_m$ generation processes

The FlexAE also uses a sequence of bits $(S_0S_1 \dots S_m)$. This sequence has the same size of the message to be sent. It is generate by applying PF_{K2} over an initialization vector (IV). The result is incremented in every 32 bits for every block of the sequence. Then the PF_{K2} is applied again to each block to generate the sequence. The IV can be random or a number that does not repeat (a NONCE).

To cipher a message, the algorithm breaks the message into m plaintext blocks

$(P_0 P_1 \ldots P_m)$. The block size (bs) in bits is $64 \times 2^{\mathrm{x}}$ bits , where $\mathrm{x} \geq 0$. The last block is padded with (10^{pb-1}), where pb is the number of padding bits to complete the block.

Each block (P_n) is XORed with the correspondent (S_n) block and it is submitted to PF_{K1} to generate a intermediate state block (st_n). The state (st_n) is submitted to PF_{K0} to generate a ciphertext block (C_{n+1}). The first ciphertext block (C_0) is a copy of the IV.

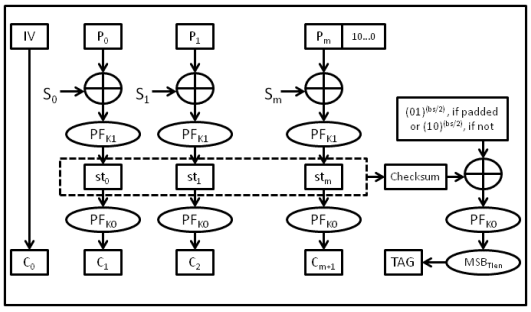

Figure 2. The FlexAE cipher

All intermediated blocks are XORed to generate a checksum. The checksum is XORed with $(10)^{bs/2}$, if the message is padded, or with $(01)^{bs/2}$ otherwise. It is submitted then to PF_{K0} to generate the TAG used for authentication. The TAG length $(Tlen)$ can be smaller than the block size, if it is adequate to the application. In this case, it truncates on its $Tlen$ more significant bits(MSB_{Tlen}).

3. Differential Cryptanalysis

The differential cryptanalysis (BIHAM and SHAMIR, 1991) technique consists on analyzing of the probabilities of the differences on the cipher SBoxes inputs and outputs.

The first step is to create a difference distribution table for each SBox. It is done calculating differences $\Delta \mathrm{X} = \mathrm{X} \oplus \mathrm{X}'$ and $\Delta \mathrm{Y} = \mathrm{Y} \oplus \mathrm{Y}'$ for every possible input pair $(\mathrm{X}, \mathrm{X}')$ and its output $(\mathrm{Y}, \mathrm{Y}')$. The $\Delta \mathrm{X}$ values are the lines and $\Delta \mathrm{Y}$ values are the columns of the table. Each cell contains the number of times that a pair $(\Delta \mathrm{X}, \Delta \mathrm{Y})$ appeared. To have a good resistance against a cryptanalysis attack, the distribution should be uniform.

The tables for FlexAE SBoxes are 256 x 256. On the the first line $(\Delta \mathrm{X} = 0)$, the only cell with a value different is $(\Delta \mathrm{X} = 0, \Delta \mathrm{Y} = 0)$ which have 256. So its probability is 1 $(p_{\Delta X=0 \to \Delta Y=0} = 1)$. For all the other lines, the maximum value is 4 this means that the maximum probability for any pair $(\Delta \mathrm{X} \neq 0, \Delta \mathrm{Y} \neq 0)$ is $p = \frac{4}{256} = 2^{-6}$.

After creating the difference distribution tables, the next step was to determine plaintext (ΩP) and ciphertext (ΩT) characteristics that are useful for the attack, these characteristics are those with $p > 0$.

By example, considering only two-rounds of the PF from the FlexAE cipher with a 64 bits block size, with the characteristics $\Omega P = 10\ 00\ 00\ 00\ 00\ 00\ 00\ 00$ and $\Omega T = 01\ 09\ 00\ 00\ 1F\ E5\ 00\ 00$, it has the probability $p^{\Omega} = 2^{-36}$. It is possible to observe that the maximum probability can be determined by the number of active SBox (that has non-zero differences on its input).

According to Heys (2001), the number of chosen plaintext pair (N_D) to perform

an attack is approximately $N_D = \frac{1}{p_D}$, where p_D is the maximum probability for the r-1 rounds of a cipher. Due to FlexAE cipher structure, there is a minimum of 2 active SBox on each round of the permutation function. For a 64 block size, the PF function is executed two times and for each time it has $(\log_2 nb + 2)$ rounds, the total number of rounds is $r = 2 \times (\log_2 8 + 2) = 10$). In this specific case $p_D = (2^{-6} \times 2^{-6})^9 = 2^{-108}$ and $N_D = \frac{1}{p_D} = 2^{108}$.

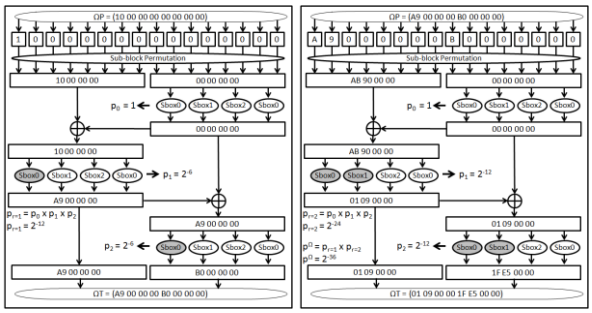

Figure 3. Calculating p^{Ω} for determined characteristics

If there are two active SBoxes on the r-1 round, it will be possible to uncover only 16 bits of the key with these chosen plaintext pairs. The maximum number of bits that can be uncovered is the same number of bits of the block size, but it will need more chosen plaintext pairs, as shown on Table 2.

Table 2. The number of choosen plaintext pairs N_D for a differential attack

Block Size	Number of key bits uncovered	Rounds (r-1)	Active SBoxes	N_D
64	64	9	26	2^{156}
128	128	11	44	2^{264}
256	256	13	78	2^{468}
512	512	15	144	2^{864}

The results on Table 2 shows that an attack using a differential cryptanalysis the the FlexAE cipher is not pratical.

4. Linear Cryptanalysis

The linear cryptanalysis (MATSUI, 1993) technique consists in evaluating the cipher using linear expressions to approximate the cipher results and calculating their biases of being true or false. The higher the bias, the easier is to uncover the key bits.

For each FlexAE SBox, there are 65025 possible linear expressions. These expressions were evaluated against all possible inputs and their biases are computed. If a linear has expression has exactly 50% chance of being true or false, it has no bias. By example, considering the input $X_7X_6X_5X_4X_3X_2X_1X_0$ and the output $Y_7Y_6Y_5Y_4Y_3Y_2Y_1Y_0$, the linear approximation for the SBox0 using the expression $X_1 \oplus X_0 \oplus Y_2 \oplus$=0 has no bias. Another approximation example for SBox0 is the expression $X_3 \oplus X_2 \oplus X_1 \oplus X_0 \oplus Y_3 \oplus Y_1 \oplus Y_0$=0, this expression has a bias $\epsilon = \frac{16}{256} = 2^{-4}$. It was determined that the maximum bias for any of the FlexAE SBoxes is $\epsilon = 2^{-4}$.

After calculating the bias for every SBox, the next step is to verify the cipher structure effect and determine the best linear expressions for each round. In this stage it

is easier to represent the linear expressions in graphic way. The Figure 4 has a graphical representation of a linear approximation for all 5 rounds of the FlexAE permutation function using 64 bits block size.

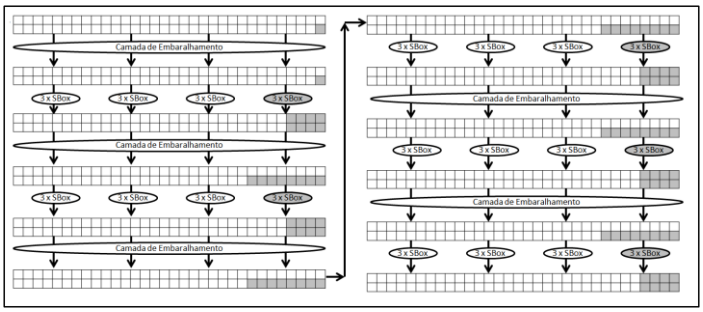

Figure 4. The linear expression graphical representation for FlexAE

The complexity of an attack is determined by the number of chosen plaintext pair (N_L) which can be calculate from the bias $N_L = \frac{1}{\epsilon^2}$ (HEYS, 2001). On the linear cryptanalysis, if the number of active SBox is known (n), the bias (ϵ) can be determined subtracting (0.5) from the probability (p) calculated using the Piling-up Lemma $\mathrm{p} = \frac{1}{2} + 2^{\mathrm{n}-1} \prod_{\mathrm{i}=1}^{\mathrm{n}} \left(\mathrm{p_i} - \frac{1}{2}\right)$ (MATSUI,1993): $\epsilon = p - 0.5$.

Table 3. The number of choosen plaintext pairs N_L for a linear attack

Block Size	Rounds (r)	Active SBox	Maximum Bias	$N_L = \frac{1}{\epsilon^2}$
64	10	30	$\epsilon = 2^{-91}$	$N_L = 2^{182}$
128	12	36	$\epsilon = 2^{-109}$	$N_L = 2^{218}$
256	14	42	$\epsilon = 2^{-127}$	$N_L = 2^{254}$
512	16	48	$\epsilon = 2^{-145}$	$N_L = 2^{290}$

Analyzing the results on Table 3, it is possible to conclude that the attack is not feasible for 64 and 128 bits block size. On these configurations the number of blocks necessary for the attack are bigger than the number of block that can be generate for the attack. For 256 and 512 bits block size, the attack is possible but it is not useful if it is as difficult as a brute force attack. In these block size configurations it is recommended to limit the key size to a maximum of 256 bits.

5. Lightweight Characteristics Comparison

The FELICS framework (DINU et al., 2015) was used to compare lightweight characteristics. It uses code size in bytes (CODE), ram memory in bytes used (RAM) and execution time in processor cycles (TIME) as metrics. To collect the data, four FlexAE implementations were created using different configurations of block and key sizes: FlexAE_64_128 (64 bits block size, 128 bits key size), FlexAE_128_256, FlexAE_256_512 and FlexAE_512_1024. The metrics were collect for two types of microcontrollers the 8-bit Atmel AVR ATmega128 and the 16-bit Texas MSP430F1611.

A rank comparing the FlexAE to other 23 implementations showed they are better than half of the implementations on the RAM and TIME metrics. So it can be considered a good candidate for lightweight cipher.

Table 4. FlexAE rank position on a comparative rank using FELICS framework

Cipher Implementation	AVR			MSP		
	Code	RAM	Time	Code	RAM	Time
FlexAE_64_128	25	1	5	25	1	10
FlexAE_128_256	27	7	13	27	10	11
FlexAE_256_512	22	22	20	26	23	14
FlexAE_512_1024	21	27	21	22	27	23

It is important to highlight: the FlexAE cipher is the only one with integrated authentication; the FlexAE_64_128 was faster than the AES cipher on both microprocessors; the FlexAE_128_256 was also faster than the AES cipher on 16-bit MSP; the FlexAE_64_128 used less RAM than all other ciphers.

6. Randomness Statistics Validation

The cipher randomness was evaluated using two processes. The first was to generate 9 dataset with different categories and submitting them to the NIST statistical test suite. This process was described by SOTO (1999) and it was used to evaluate AES candidates. To maintain compatibility to the datasets, only a implementation with 128 bits block size and 256 bits key size could be used.

The other process was to encrypt a 64 bits counter with a key and submit it to the dieharder tool (BROWN, 2016). On this process the implementations FlexAE_64_128, FlexAE_128_256, FlexAE_256_512 and FlexAE_512_1024 were tested.

The tests had shown the data generated by the algorithm implementations cannot be distinguished from a random sequence, confirming the algorithm randomness.

7. Conclusion

This works presented the FlexAE cipher. It is a lightweight cipher with integrated authentication. The analysis had indicates the cipher appears to have a good resistance against differential and linear cryptanalysis attacks. The cipher was compared to other lightweight cipher using the FELICS framework. It presented similar metrics to other ciphers. The tests also had shown that the cipher has a good randomness.

This work leaves space for future researches like: the cipher modification to improve its resistance against linear cryptanalysis; compare the FlexAE to the CEASAR Competition finalists; a hardware implementation of cipher can be built to compare it to other ciphers in hardware.

References

BEAULIEU, R. et al. SIMON and SPECK: Block Ciphers for the Internet of Things, 2015. URL: <http://eprint.iacr.org/2015/585>. Access Date: Oct 1st 2016.

BEIERLE, C. et al. The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS. Advances in Cryptology -- CRYPTO 2016: 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II, Berlin, Heidelberg, 2016. 123-153.

BIHAM, E.; SHAMIR, A. Differential cryptanalysis of DES-like cryptosystems. Journal of CRYPTOLOGY, 4, n. 1, 1991. 3-72.

BORGHOFF, J. et al. PRINCE-a low-latency block cipher for pervasive computing applications. Advances in Cryptology-ASIACRYPT 2012, 2012. 208-225.

BROWN, R. G.; EDDELBUETTEL, D.; BAUER, D. Dieharder: A Random Number Test Suite, 2016. URL: <http://phy.duke.edu/~rgb/General/dieharder.php>. Access Date: May 13th 2016.

CRYPTOGRAPHIC Competitions. CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness, 2016. URL: <https://competitions.cr.yp.to/caesar.html>. Access Date: Mar 22nd 2017.

CRYPTOLUX RESEARCH GROUP - UNIVERSITY OF LUXEMBOURG. Lightweight Block Ciphers, 2016. URL: <https://www.cryptolux.org/index.php/Lightweight_Block_Ciphers>. Access Date: Oct 1st 2016.

DAEMEN, J.; RIJMEN, V. Specification for the advanced encryption standard (AES). Federal Information Processing Standards Publication, 2001.

DINU, D. et al. FELICS – Fair Evaluation of Lightweight Cryptographic Systems, jul. 2015. URL: <http://csrc.nist.gov/groups/ST/lwc-workshop2015/papers/session7-dinu-paper.pdf>. Access Date: Oct 12th 2016.

EVEN, S.; MANSOUR, Y. A construction of a cipher from a single pseudorandom permutation. Journal of Cryptology, 10, 1997. 151-161.

GUO, J. et al. The LED block cipher. Cryptographic Hardware and Embedded Systems-CHES 2011, 2011. 326-341.

ISO/IEC 29192-1:2012. Information technology - Security techniques - Lightweight cryptography - Part 1: General. Geneva: ISO, 2012.

JUTLA, C. S. Encryption modes with almost free message integrity. International Conference on the Theory and Applications of Cryptographic Techniques, 2001. 529-544.

MATSUI, M. Linear cryptanalysis method for DES cipher. Workshop on the Theory and Application of of Cryptographic Techniques, 1993. 386-397.

Nascimento, E. M.; Xexéo, J.A.M. "A flexible authenticated lightweight cipher using Even-Mansour construction". 2017 IEEE International Conference on Communications (ICC), Paris, 2017, pp. 1-6. (doi: 10.1109/ICC.2017.7996734). URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7996734&isnumber=7996317>. Access Date: Jun 17th 2018.

Nascimento, E.M. “Algoritmo de Criptografia Leve com Utilização de Autenticação”. 2017. 113p. Dissertação (mestrado) - Instituto Militar de Engenharia, Rio de Janeiro, 2017. URL: <http://www.comp.ime.eb.br/pos/arquivos/publicacoes/dissertacoes/2017/2017-Eduardo.pdf>. Access Date: Jun 23rd 2018.

SOTO, J. Randomness testing of the AES candidate algorithms, 1999. URL: < http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.39.231&rep=rep1&type=pdf >. Access Date: Jun 23rd 2018.