Some evaluations for involutory diffusion layer of 64-bit AES-like block ciphers based on the hadamard matrices

Tran Duy Lai, Nguyen Van Long, Nguyen Bui Cuong

Tóm tắt— **Trong bài báo này, chúng tôi phân loại và đánh giá các tầng khuếch tán cuộn của mã khối 64-bit tựa AES dựa trên các ma trận Hadamard. Đầu tiên, chúng tôi tính toán số lượng các điểm cố định trong lớp này. Sau đó, chúng tôi thực hiện một cuộc khảo sát thực tế trên các ma trận Hadamard 4×4-bit trong GF(2^4) và thay đổi các số cổng XOR được sử dụng để cài đặt như mô tả trong [9] nhằm chọn ra một ma trận phù hợp để xây dựng các tầng khuếch tán cuộn của mã khối tựa AES với kích thước khối 64-bit.**

Abstract— **In this paper, we classify and evaluate the involutory diffusion layer of 64-bit AES-like block ciphers based on the Hadamard matrices. Firstly, we calculate the number of fixed points in this layer. After that, we perform a practical investigation on the 4×4-bit Hadamard matrices in GF(2^4) and modify the number of XOR gates used to implement as described in [9] to choose a suitable matrix for constructing the involutory diffusion layer of AES-like block ciphers with 64-bit block size.**

Keywords*— *involutory diffusion layer; Hadamard matrix; fixed point; XOR Count.

I. INTRODUCTION

Since 2000, the standardization of Rijndael being as the Advanced Encryption Standard (AES), there are a surprisingly large number of new primitives which have the similar components as in the AES block cipher [1]. For example, some AES-like block ciphers such as ANUBIS [12], LED [4], PRINCE [13],… also, some hash functions as PHOTON [5], GOST R 34.11.2012 [6]... This is mostly because applied *the wide trail strategy*. This strategy is not only guarantee good diffusion properties but also allow the designers to easily give a security bound against the differential and linear cryptanalysis [11]. Moreover, as another advantage, this split the chosen of the linear layer and the non-linear one separately. For AES-like block ciphers, (all as above), the diffusion layer includes a linear transformation such as *MixColumns* and some as *ShiftRows* in AES. For this layer, some secure criteria about the diffusion property have been proposed. In particular, the essential requirement is these transformation have a high branch number (see [1]). Moreover, another criterion is fixed points which are exploited in some recently effective attacks (see [2]). Finally, an important requirement is the hardware-implementation ability of the diffusion layer in these ciphers, specially in 64-bit block ciphers tailored for implementation in constrained environments.

Related works. In [16], we have considered a general model of diffusion which using an alternative transformation instead of *ShiftRows* operator for a 64-bit lightweight block cipher. However, we have not proposed choosing a particular MDS matrix for this *MixColumns* operator.

The authors of [7] have proposed a method for constructing the involutory MDS matrices based on two Vandermond matrices. Another method for building a MDS matrix with an arbitrary size, which is a form as the Cauchy matrix, has been given in [8,10]. These obtained matrices are the Hadamard matrices. By the most recently research of S.M.Sim *et* al. in [9] about the involutory MDS Hadamard matrices using in the lightweight designs, they are only interested in the branch number of the obtained diffusion layer and their hardware-implementation ability but not about fixed point property.

Our Contribution. In this paper, based on the proposed model which using a transformation like *ShiftRows* operator, we consider of finding an involutory MDS Hadamard matrix using a transformation like *MixColumns* to construct a secure involutory diffusion layer effectively. First, we consider the cryptographic criteria (branch numbers, fixed points) and perform both a searching for all 4×4-bit MDS Hadamard matrices in $\mathbb{F}_2^4$ and a more precisely calculation of number of XOR count than in [9]. Then, we consider and analyze the best implemented matrix of this form.

Outline. Our paper is structured as follow. In Section 2, we introduce some related notions. Next, in Section 3 we propose a model of diffusion which has AES-like properties and recall some results of the general model, also applies method of [2] to implies the number of fixed points for the diffusion layer using the involutory Hadamard matrices. Finally, In Section 4, we classify the Hadamard matrices of size 4×4 in $\mathbb{F}_2^4$ and give some implemented evaluations of the involutory Hadamard matrices.

II. PRELIMINARIES

A. Notions

Let $\mathbb{F}_{2^r}$ be a finite field of 2^r elements, $r\geq 1$. Since $\mathbb{F}_{2^r}$ be isomorphic to polynomial in $\mathbb{F}_2[X]$ which is reduced by an irreducible polynomial *p*(*X*) of degree *r*, i.e, each element in $\mathbb{F}_{2^r}$ can be considered as a polynomial *a*(*X*) of degree *r* –1 with coefficients in $\mathbb{F}_2$ as follow: $a(X)=\sum_{i=0}^{r-1} b_i X_i, b_i\in\mathbb{F}_2,\quad 0\leq i\leq r-1$. We can also consider a(X) as a sequence of *r* bit ($b_{r-1},\ldots,b_0$). In the remains of this paper, an element α in $\mathbb{F}_{2^r}$ can be seen as a polynomial a(X) or a r-bit binary sequence. The addition on $\mathbb{F}_{2^r}$ is simply defined as the exclusive-OR (XOR) operation of coefficients of polynomials with respect to each elements, which does not depend on the choice of irreducible polynomial *p*(*X*). However, we need to determine the irreducible polynomial of degree *r* when we work with the multiplications. In this case, we define this field as $\mathbb{F}_{2^r}/p(X)$.

B. The MDS and Hadamard matrices

The Maximal Distance Separable matrices play important role in cryptography designs due to it guarantees a perfect diffusion layer. In this section, we recall some notions, properties of MDS matrices. We denote I_k be an unit matrix of size $k\times k$.

Definition 1 ([9]). *The branch number of a* $k\times k$ *matrix M on* $\mathbb{F}_{2^r}$ *is the minimum number of non-zero components of input vector v and output vector u = v.M (denoted by wt(v) and wt(u), respectively) for all* $v\in(\mathbb{F}_{2^r})^k/\{0\}$.*That means, if the branch number equals to* $\min_{v\neq 0}\{\mathrm{wt}(v)+\mathrm{wt}(u)\}$ *and the optimal values k+1 obtains, then we say that M is a MDS matrix.*

Definition 2 ([9]). *A finite field Hadamard (or simply called Hadamard) matrix H is a* $k\times k$ *matrix, with k =* 2^s*, that can be represented by two other submatrices* H_1 *and* H_2 *which are also Hadamard matrices:*

$$H=\begin{pmatrix} H_1 & H_2 \\ H_2 & H_1 \end{pmatrix}$$

As in [9], we denoted by *had*($h_0, h_1,\ldots,h_{k-1}$) a Hadamard matrix (with $h_i = H_{0,i}$ standing for the entries of the first row of the matrix), where $H_{i,j} = h_{i\oplus j}$ and $k=2^s$. By multiplying directly, we see that if *H*=*had*($h_0,h_1,\ldots,h_{k-1}$) is a Hadamard matrix then

$$H\times H=c^2\cdot I_k$$

where $c^2=h_0^2+h_1^2+\ldots+h_{k-1}^2$. In other words, the product of a Hadamard matrix with itself is a multiple of an identity matrix, where the multiple c^2 is the sum of the square of the elements from the first row. Thus, an important and direct subsequence is that a Hadamard matrix H is involutory on $\mathbb{F}_{2^r}$ if the sum $h_0\oplus h_1\oplus\ldots\oplus h_{k-1}$ equal to 1.

C. Evaluate the number of XOR operations used to implement the matrix multiplication

In this section, we represent a notion, XOR count, that we will use as a measure to evaluate the lightweightness of a given matrix. The XOR count depends on the finite field and irreducible polynomial which is considered. Evaluations based on XOR count are new line in implement evaluating some cryptography components and exploited in many recent research. As described in [15], the low XOR count will be related to the minimization of implement area. In order to compute implement source of the *MixColumns* layer, we often convert it to considering the necessary XOR count for the multiplication of matrix M which represent *MixColumns*. Moreover, the number of multiplications when multiply one row of M by one column of an arbitrary state matrix equal to $\sum_{i=1}^{k}\gamma_i+(n-1)\times r$ (see [15] for detail) where γ_i be the XOR count of the i^{th} element in a row of matrix M, n be the number of non-zero number component in this row and r be the dimension of the finite field. Then, in order to calculate the XOR count to perform the matrix multiplication, we need to find the total XOR counts which are implemented for all rows of the matrix. But in some recent block ciphers, the matrices which represent the diffusion layer are often chosen such that we just implement only one row of the matrix but we can perform computations of output values of the matrix following the strategy based on companion matrices in [3] or Hadamard matrices which we were considering in this paper.

III. THE INVOLUTORY DIFFUSION LAYER OF THE 64-BIT AES-LIKE BLOCK CUPHERS BASED ON THE HADAMARD MATRICES

A. Promoted 64-bit diffusion layer mode

In [16], we have proposed a block cipher model which has a SPN structure similar as 64-bit AES, where the diffusion layer includes a *TranCells* transformation, is a cell transposition (cell is a nibble or 4-bit string) (see Figure.1), and a *MixColumns* transformation based on a general MDS matrix as follow:

We have proved that this *model* is secure against the differential and linear cryptanalysis similar as AES by the number of active S-boxes as follow:

Proposition 1. (Proposition 2, [16]) *The minimal number of active cell in four arbitrary consecutive rounds equals to 25.*

In this paper, we consider *MixColumns* transformations based on the involutory MDS matrices of size 4×4 in $\mathbb{F}_{2^4}$ has the form *Had*(a_0, a_1, a_2, a_3) where $a_i \in \mathbb{F}_2^4 \setminus \{0\}$.

B. The number of fixed points in the promoted diffusion layer based on the Hadamard matrix

In our model, we can represent the diffusion layer as a multiplication of matrix A of size 16×16 by a column vector of size 16 when using both *TranCells* and *MixColumns* transformations. The number of fixed points in this 64-bit diffusion layer equal to exact the solutions of equation $(A \oplus I)X = 0$, where I be the unit matrix of size 16×16, $X = (x_0, x_1, ..., x_{15}), x_i \in \mathbb{F}_2^4, i = 0,1,...,15$. More deep analysis about fixed points can be seen in [2].

Fact 1. *The TranCells transformation does not effect the number of fixed points in the diffusion layer which includes it.*

Therefore, the number of fixed points in the proposed diffusion layer model equal to the number of fixed points of the *MixColumns* transformation. The essence of the *MixColumns* transformation is that performing multiply matrix 4×4 M by 4 distinct columns of a state matrix 4 times. Assume that *L** be the transformation has representative matrix *M*. Then,

Fact 2. *If the transformation L* has N_{L^*} fixed points, then MixColumns has $N_{L^*}^4$ fixed points, that is exact fixed points of the diffusion layer.*

Proposition 2. *The diffusion layer which includes 2 transformations TranCells and MixColumns, where the MixColumns transformation using the involutory MDS Hadamard matrix of size 4×4 in $\mathbb{F}_{2^4}$, has 2^{32} fixed points.*

Proof: We consider *A* is a involutory MDS Hadamard matrix of size 4×4 in $\mathbb{F}_{2^4}$ represented as *A*= *had*(a_0,a_1,a_2,a_3), where $a_i \in \mathbb{F}_2^4 \setminus \{0\}$ and $a_0 \oplus a_1 \oplus a_2 \oplus a_3 = 1$ i.e, $a_0 \oplus 1 = a_1 \oplus a_2 \oplus a_3$. Then the number of fixed points on the proposed diffusion layer which use matrix A in the *MixColumns* transformation is $N_L = 2^{4\cdot 4\cdot(4-rank(A\oplus I))}$. Let's consider $A \oplus I$, where I be the unit matrix of size 4×4, we have:

$$\text{rank}(A \oplus I) = \text{rank}\begin{bmatrix} a_1 \oplus a_2 \oplus a_3 & a_1 & a_2 & a_3 \\ a_1 & a_1 \oplus a_2 \oplus a_3 & a_3 & a_2 \\ a_2 & a_3 & a_1 \oplus a_2 \oplus a_3 & a_1 \\ a_3 & a_2 & a_1 & a_1 \oplus a_2 \oplus a_3 \end{bmatrix} = 2$$

Therefore, we have $N_L = 2^{4\cdot 4(4-2)} = 2^{32}$ □

Our analysis research show that if we use the involutory MDS Hadamard matrices in the *MixColumns* transformation as in the considered diffusion layer model then there exist many fixed points.

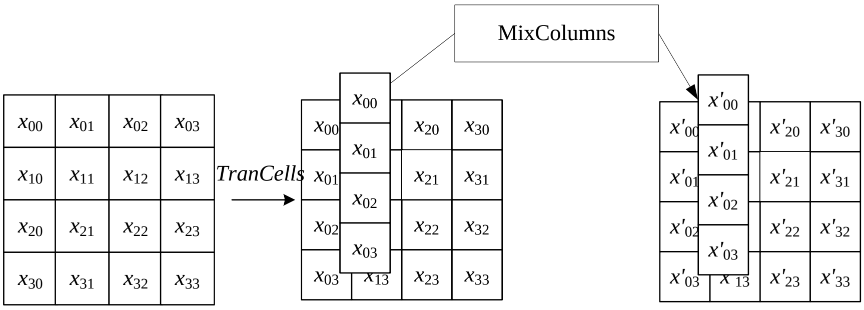
MixColumns
TranCells
x_{00} x_{01} x_{02} x_{03}
x_{10} x_{11} x_{12} x_{13}
x_{20} x_{21} x_{22} x_{23}
x_{30} x_{31} x_{32} x_{33}
x'_{00} x'_{01} x'_{02} x'_{03}
x'_{13} x'_{20} x'_{21} x'_{22} x'_{23}
x'_{30} x'_{31} x'_{32} x'_{33}

Figure 1. The proposed diffusion layer model

IV. CLASSIFICATION OF THE 4X4 MDS HADAMARD MATRICES IN $\mathbb{F}_4$

In the following of this paper, we will find all 4×4 MDS Hadamard matrices in $\mathbb{F}_{2^4}$ to choose the best involutory matrix for designing the 64-bit diffusion layer by the proposed model.

A. Equivalence of the Hadamard matrices

In order to classify the Hadamard matrices, we consider an following equivalent relation:

Definition 4. *Given 2 $k\times k$ Hadamard matrices H=had(h_0,h_1,...,h_{k-1}) and H'=had($h'_0,h'_1,...,h'_{k-1}$), then it is said that H and H' are equivalent relation if there exist a permutation σ of k elements such that $h'_i = h_{\sigma(i)}$. Denoted by H~H'.*

To classify the MDS Hadamard matrices by our criteria, we use the following properties:

Fact 3. *Given two equivalent Hadamard matrices H~H'. Then, H and H' have the same the number of branch, fixed points, implement XOR count and involutory property.*

Thus, based on this property we can reduce the complexity of considering the number of Hadamard matrices by just consider the representative elements (arrange in order) in each permutation equivalent class because all elements in the same equivalent class have the same cryptography and implementation properties.

B. Our results

We can perform an exhausted search for all possibility of (a_0,a_1,a_2,a_3) in $\mathbb{F}_{2^4}$, then using the algorithm described in [14] to check the MDS property for all matrices. The number of matrices need to check is 2^{16}. However, by the equivalent correlation we just consider all the representative elements implies that the complexity will be reduced. The experiment results show that among 22680 MDS matrices (945 representatives) there are 1512 involutory matrices (63 representatives), i.e, 4-tuple (a_0,a_1,a_2,a_3) arranged satisfying $\bigoplus_{0\le i\le3} a_i = 1$. After combining with the *TranCells* transformation to obtain the 64-bit diffusion layers as our proposed model, we calculate the number of fixed points of all diffusion layers. In detail, there are 21168 (number of remains which does not have involutory property) MDS matrices which generate the diffusion layer with none of fixed points. For hardware evaluations, in [9] the authors also evaluated the hardware implement ability of the involutory MDS Hadamard matrices based on the hardware implement basic for each multiplication of elements in the finite field. In $\mathbb{F}_{2^4}$, authors give the evaluation of necessary XOR count to multiply by 16 elements in $\mathbb{F}_{2^4}$ when using polynomial generator as a primitive polynomial $f(x)=x^4+x+1$. The XOR count in Table 1:

TABLE 1. XOR COUNTS WHEN MULTIPLY BY EACH ELEMENTS IN $\mathbb{F}_{2^4}$

Elements	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
[9]	0	0	1	5	2	6	5	9	3	1	8	6	5	3	8	6
This paper	0	0	1	4	2	4	5	6	3	1	5	4	4	2	5	4

Actually, the evaluation in [9] is not tight due to the repeat of sum of some variables but in our evaluation, we have modified. For example, the multiplication $Y = X \cdot 7$ in $\mathbb{F}_{2^4}$ can be done as follow:

$$Y = X \cdot 7 \Leftrightarrow$$

$$\begin{cases} y_0 = x_0 \oplus x_2 \oplus x_3, & y_1 = x_0 \oplus x_1 \oplus x_2 \\ y_2 = x_0 \oplus x_1 \oplus x_2 \oplus x_3, & y_3 = x_1 \oplus x_2 \oplus x_3 \end{cases} \quad (1)$$

where $Y = (y_3 \| y_2 \| y_1 \| y_0)$, $X = (x_3 \| x_2 \| x_1 \| x_0)$, $y_i, x_i \in \mathbb{F}_2$. If we construct hardware circle for each Boolean function in the system equations (1) then the required XOR counts is 9 (as in the evaluation in [9]). Actually, we have:

$$(1) \Leftrightarrow \begin{cases} u = x_2 \oplus x_3,\ v = x_0 \oplus x_1 \\ y_0 = x_0 \oplus u,\ y_1 = v \oplus x_2 \\ y_2 = v \oplus u,\ y_3 = x_1 \oplus u \end{cases}$$

This system requires only 6 XOR counts. Doing the same with other components, we get the XOR count evaluation as in Table 2.

For all 22680 MDS Hadamard matrices in $\mathbb{F}_{2^4}$, we evaluate the necessary XOR count for both 4×4 MDS matrix and its inverse. The evaluation is total number of XOR operators when multiply one column vector with 4 nibbles by one row of matrix $Had(a_0, a_1, a_2, a_3), a_i \in \mathbb{F}_{2^4}, 0 \le i \le 3$ and one row of its inverse.

Table 2 results in ascending necessary XOR counts for hardware implementation of all these 4×4 MDS Hadamard matrices, where "+" denote the involutory matrices and "-" for non-involutory matrices. Therefore, among 22680 involutory MDS Hadamard matrices $Had(1,4,9,13)$ is a representative of class 1 in Table 2 has the best implementation with necessary cryptographic properties.

C. Analyse the Had (1,4,9,13) matrix

In this section, we evaluate total necessary XOR counts with involutory MDS Hadamard matrix *Had* (1,4,9,13). The evaluation is total number of XOR operators when multiply one column vector with 4 nibbles by one row of matrix *Had* (1,4,9,13), result of this transformation is a 4 bits string, denoted by R. So, R mapping from $\mathbb{F}_2^{16}$ to $\mathbb{F}_2^4$. Indeed, we consider this transformation R because all rows in Hadamard matrix have the same elements. In order to multiply one column vector with 4 elements by the Hadamard matrix, we just need to use a multiplication diagram and permute input nibbles then we get the output column vector. This is advantages of the Hadamard matrix. By this analysis, we can need only a multiplication diagram based on transformation R to multiply two 4×4 matrices in full *MixColumns* transformation.

TABLE 2. CLASSIFICATION OF 22680 MDS HADAMARD MATRICES OF SIZE 4×4 IN $\mathbb{F}_{2^4}$ WITH GENERAL POLYNOMIAL $f(x)=x^4 \oplus x \oplus 1$

Class	XOR count	Involution	No. of fixed points	No. of MDS Hadamard matrices	Class	XOR count	Involution	No. of fixed points	No. of MDS Hadamard matrices
1	17	+	2^{32}	24	**11**	40	-	1	144
2	18	+	2^{32}	24	**12**	41	-	1	192
3	20	+	2^{32}	24	**13**	42	-	1	192
4	21	+	2^{32}	96	**14**	43	-	1	240
5	22	+	2^{32}	192	**15**	44	-	1	480
6	23	+	2^{32}	120	**16**	45	-	1	816
7	24	+	2^{32}	120	**17**	46	-	1	1104
8	[25…31]	+	2^{32}	912	**18**	47	-	1	1104
9	37	-	1	48	**19**	48	-	1	1728
10	38	-	1	96	**20**	[49…64]	-	1	15024
Total: 22680									

First, we consider the following illustrative multiplication diagram for 4, 9 and 13 as follow:

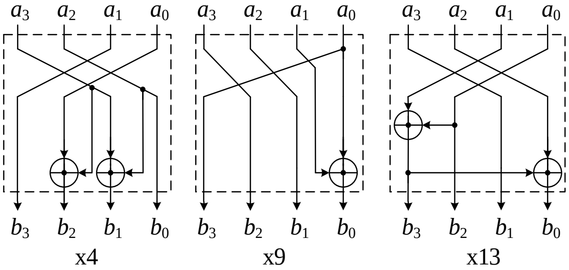

Figure 2. The multiplication for 4, 9 and 13 in $\mathbb{F}_{2^4}$ with generator polynomial $x^4 \oplus x \oplus 1$

Therefore, the transformation of matrix *Had* (1,4,9,13) for an input column $(a,b,c,d) \in \mathbb{F}_2^{16}$, where $a = (a_3, a_2, a_1, a_0), ..., d = (d_3, d_2, d_1, d_0)$ has respectively output $x \in \mathbb{F}_2^4$, $x = 1 \cdot a \oplus 4 \cdot b \oplus 9 \cdot c \oplus 13 \cdot d$ as the following illustration:

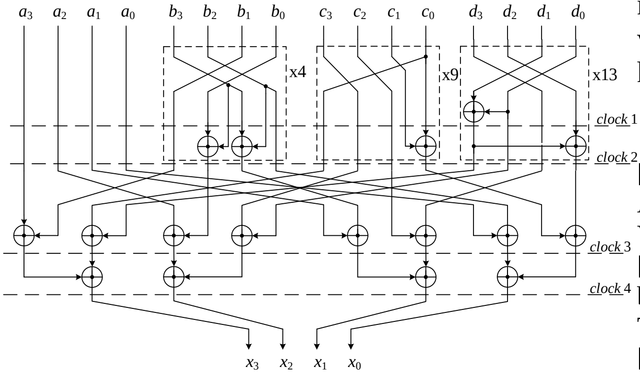
a_3 a_2 a_1 a_0 b_3 b_2 b_1 b_0 c_3 c_2 c_1 c_0 d_3 d_2 d_1 d_0
x4
x9
x13
clock 1
clock 2
clock 3
clock 4
x_3 x_2 x_1 x_0

Figure 3. Hardware design illustration of R transformation

Thus, we need 17 XOR counts and 4 clock cycle to implement R transformation. Moreover, in order to perform a full *MixColumns* transformation based on the multiplication diagram on Figure 3, we need to apply this diagram 16 times with also 17 XOR count but the clock cycle will be 16×4 = 64. However, we are interested in the limited source devices in this design. So, the necessary XOR count should be more interested. Since the matrix is involutory MDS Hadamard then the diagram on Figure 3 can be used for both encryption and decryption. To compare the complexity with real designs, we also construct the multiplication diagram for R' and (R')$^{-1}$ when multiply companion matrix *Companion* (2,2,1,4) by its inverse which used in the *MixColumnSerial* transformation of lightweight block cipher LED has the same as an column vector [4] (Table 3, 1 XOR count value ~ 2,65 GE).

TABLE 3. COMPLEXITY COMPARISON IN HARDWARE IMPLEMENTATION OF TWO MODEL

Transformation	**XOR count**	**Equi. GE value**	**Clock cycle**
MixColumns based on Had(1,4,9,13)	17	45,05	64
MixColumnSerials in LED	14	37,10	48
InvMixColumnSerials in LED	18	47,70	64

V. CONCLUSION

In this paper, we have proposed and evaluated the involutory diffusion layer based on the Hadamard matrices in AES-like block ciphers. The experiment results show that which ciphers have the diffusion layer using involutory MDS Hadamard matrices will be advantage in implementation and secured against differential and linear cryptanalysis as in the AES cipher model by the wide trail strategy. However, they will hide some weakness due to the diffusion layer has many fixed points.

REFERENCES

[1]. Daemen J, Rijmen V, “The Design of Rijndael, AES” - The Advanced Encryption Standard. Springer-Verlag, 2002.

[2]. Z'aba MR, “Analysis of linear relationships in block ciphers”. Ph.D. Thesis, Queensland University of Technology, Brisbane, Australia, 2010.

[3]. Kishan Chand Gupta and Indranil Ghosh Ray, “On Constructions of MDS Matrices form Companion Matrices for Lightweight Cryptography”. Security Engineering and Intelligence Informatics, volume 8128, pp. 29-43, 2013.

[4]. Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew J. B. Robshaw, “The LED block cipher”. In CHES 2011, vol. 6917, pp. 326-341, 2011.

[5]. Jian Guo, Thomas Peyrin and Axel Poschmann, “The PHOTON Family of lightweight Hash Functions”. In CRYPTO, Springer, pp. 222-239, 2011.

[6]. ГОСТ Р 34.11-2012. “Криптографическая защита информации” – Функция хэширования. Издание официальное, Москва 2012.

[7]. M. Sajadieh, M. Dakhilalian, H. Mala, B. Omoomi, “On construction of involutory MDS matrices from Vandermonde Matrices in GF(2^q)”. Design, Codes Cryptography, pp. 1-22, 2012.

[8]. Youssef, A. M., Mister, S., Tavares, S.E, “On the Design of Linear Transformations for Substitution Permutation Encryption Networks”. In: Workshop on Selected Areas in Cryptography, SAC 1997, pp. 40-48.

[9]. Sim, Siang Meng, et al, "Lightweight MDS Involution Matrices." Fast Software Encryption (FSE), 2015.

[10]. Kishan Chand Gupta and Indranil Ghosh Ray. “On Constructions of Involutory MDS Matrices”. AFRICACRYPT 2013, LNCS 7818, pp. 43-60, 2013.

[11]. Joan Daemen, Vincent Rijmen, "The Wide Trail Design Strategy". IMA Int. Conf, pp. 222-238, 2001.

[12]. Brumley, Billy Bob, "Secure and fast implementations of two involution ciphers". Information Security Technology for Applications. Springer Berlin Heidelberg, pp. 269-282, 2012.

[13]. Borghoff, Julia, et al, "PRINCE–a low-latency block cipher for pervasive computing applications". Advances in Cryptology–ASIACRYPT 2012. Springer Berlin Heidelberg, pp. 208-225, 2012.

[14]. Kishan Chand Gupta and Indranil Ghosh Ray, "On Constructions of MDS Matrices form Companion Matrices for Lightweight Cryptography". Security Engineering and Intelligence Informatics, vol. 8128, pp. 29-43, 2013.

[15]. Khoo, Khoongming, et al, "FOAM: Sear-ching for Hardware-Optimal SPN Structures and Components with a Fair Comparison". Cryptographic Hardware and Embedded Systems-CHES 2014. Springer Berlin Heidelberg, pp. 433-450, 2014.

[16]. Nguyen Van Long, Tran Duy Lai, Nguyen Bui Cuong. "Some evaluations for a replacement of shitfrows operator in AES-like ciphers", Journal of Science and Technology Military, 2015 (Vietnamese).

SƠ LƯỢC VỀ TÁC GIẢ

TS. Trần Duy Lai

Đơn vị công tác: Viện Khoa học - Công nghệ Mật mã, Ban Cơ yếu Chính phủ, Hà Nội.

Email: tdlai@bcy.gov.vn

Tốt nghiệp chuyên ngành Xác suất - Thống kê, Đại học Matxcơva năm 1985. Nhận bằng Tiến sĩ ngành Toán ứng dụng, Đại học Bách khoa Hà Nội năm 1996.

Hướng nghiên cứu hiện nay: Toán, Khoa học - Công nghệ Mật mã, Bảo mật thông tin trên mạng máy tính.

TS. Nguyễn Văn Long

Đơn vị công tác: Viện Khoa học -Công nghệ Mật mã, Ban Cơ yếu Chính phủ, Hà Nội.

E-mail: nvlong.bcy@gmail.com.

Tốt nghiệp chuyên nghành An toàn thông tin các Hệ thống viễn thông, năm 2008 và nhận bằng Tiến sĩ chuyên ngành Các phương pháp bảo vệ thông tin, Học viện FSO, Liên bang Nga năm 2015.

Hướng nghiên cứu hiện nay: Khoa học mật mã, mã hóa đối xứng.

ThS. Nguyễn Bùi Cương

Đơn vị công tác: Viện Khoa học - Công nghệ Mật mã, Ban Cơ yếu Chính phủ, Hà Nội.

E-mail: nguyenbuicuong@gmail.com

Tốt nghiệp chuyên ngành Toán học, Đại học Sư phạm Hà Nội - Đại học Quốc gia Hà Nội năm 2004. Tốt nghiệp Thạc sĩ Toán học, Đại học Khoa học Tự nhiên - Đại học Quốc gia Hà Nội năm 2008.

Hướng nghiên cứu hiện nay: Khoa học mật mã, mã hóa đối xứng.