Mathematical Problems of Computer Science 33, 121–126, 2010.

Linear Cryptanalysis of Block Ciphers in the Cluster Computational Environment

Melsik Kyuregyan, Ofelya Manukyan and Edita Harutyunyan

Institute for Informatics and Automation Problems of NAS of RA
e-mail: melsik@ipia.sci.am, manofa81@yahoo.com

Abstract

This paper presents some results concerning synthesis of new cryptosystems equivalent to SAFER+ and SAFER++ to perform their linear cryptanalysis in the cluster computational environment. A parallel software package "LinearCryptanalyser" is developed to find such "Armenian Shuffles" which were chosen as secure against differential cryptanalysis and now will be checked if they are also secure against linear cryptanalysis. The research is focused on both theoretical and practical aspects of existence of linked I/O sums. The software package "LinearCryptanalyser" analyzes the existence of linked I/O sums and the absence of such sums will indicate cryptoresistance of block ciphers against last-round attack.

1. Introduction

Generally, cryptanalysis of two types: differential and linear cryptanalysis are carried out to check security of block ciphers. The research is mainly aimed to synthesize new block ciphers of SAFER family that would be equivalent to SAFER+ and SAFER++ block ciphers and would offer substantial improvement over the previous ciphers in SAFER family from viewpoint of speed. To this aim parallel software packages were designed to perform differential and linear cryptanalysis of the new block ciphers in the cluster computational environment. These packages allow the users to examine a great many coordinate permutations "Armenian Shuffle" and to perform differential and linear cryptanalysis of block ciphers corresponding to these permutations. The new block cryptosystems, chosen as strongly secure against differential cryptanalysis, are also subjected to linear cryptanalysis. Thus, all the block ciphers that are secure against differential and linear cryptanalysis could be used for data coding as secure and fast block ciphers. The current research was carried out in statistical aspects of linear cryptanalysis and a parallel software package "LinearCryptanalyser" is developed to find such "Armenian Shuffles" which will allow construction of block ciphers that are secure against last-round attack. Efficient algorithms investigated by the authors of this paper were implemented for finding effective homomorphic I/O sums. First, a function is implemented to create and analyze the dependencies table between certain bits of the PHT output vector on certain bits of the PHT input vector. Secondly, a function is implemented to create the database of effective

homomorphic I/O sums for the PHT-function. The software package "LinearCryptanalyser" analyzes the existence of
I/O sums and the absence of such sums will indicate cryptoresistance of block ciphers against last-round attack. Current research is focused on both theoretical and practical aspects of existence
of linked I/O sums with large bias within 6 rounds.

2. Definitions and Implementation

The linear attack explores linear relations between plaintext, ciphertext and subkey bits. Linear approximations for an iterated cipher are usually made by combining approximations for each round. Let's introduce the details of the statistical research.

If $X_i = (x_n, x_{n-1}, \ldots, x_2, x_1)$ is an n-bit input to a round, $R(X_i)$ is its output, and K_i is the round subkey, then a linear relation can be expressed as

$$X_i \cdot \Gamma I \oplus R(X_i) \cdot \Gamma O = K_i \cdot \Gamma K_i \qquad (1)$$

where ΓI,ΓO and ΓK_i are n-bit masks which specify the bits of X_i, $R(X_i)$ and K_i involved in the linear relation. For example, $X_i \cdot \Gamma I = X \cdot 45_x = x_1 \oplus x_3 \oplus x_7$ (the subscript 'x' indicates hexadecimal values). The left-hand side of equation (1) provides an estimate for the XOR of the subkey bits on the right-hand side. Without loss of generality, the following simplified equation is employed

$$X_i \cdot \Gamma I \oplus R(X_i) \cdot \Gamma O = 0 \qquad (2)$$

Two numerical values can be associated with (2): first is the probability $p = \frac{\Pr(X_i \cdot \Gamma I = R(X_i) \cdot \Gamma O))}{2^n}$, that expresses the frequency with which equation (2) holds (relation (2) is also called a linear approximation). Second, the deviation of parity of (2) from a random relation, or $p' = p - \frac{1}{2}$. It is clear that $-\frac{1}{2} \le p' \le \frac{1}{2}$ and the approximation is useful only if $p' \neq 0$. The absolute value $\varepsilon = p'$ is called bias. The larger the bias the more useful the linear relation is, that is, the more unbalanced the parity of (2) from a random distribution, the less plaintext is needed to estimate the value of $K_i \cdot \Gamma K_i$. The number N of known plaintexts required for an attack using a linear relation with bias ε equals $N = c \cdot \varepsilon^{-2}$, where c is a small constant, which depends on the algorithm used for the estimation. In case $p' < 0$, the value obtained for $K_i \cdot \Gamma K_i$ is actually $\overline{K_i \cdot \Gamma K_i} = (K_i \cdot \Gamma K_i) \oplus 1$. The following notation will be used to represent a binary-valued linear relation for one round of an iterated (n-bit block) cipher:

$$\Gamma = (\Gamma I, \Gamma O, \varepsilon) \qquad (3)$$

One-round linear relations can be concatenated or stacked in order to approximate more rounds. If $\Gamma_1 = (\Gamma X_1, \Gamma Y_1, \varepsilon_1)$, $\Gamma_2 = (\Gamma X_2, \Gamma Y_2, \varepsilon_2)$ are r_1-round and r_2-round independent linear relations, respectively, and $\Gamma Y_1 = \Gamma X_2$, then it is possible to combine them to form an $(r_1 + r_2)$-round linear relation $\Gamma_3 = (\Gamma X_1, \Gamma Y_2, \varepsilon)$ with bias $\varepsilon = 2 \cdot \varepsilon_1 \cdot \varepsilon_2$.

Let's now define I/O sum $S^{(i)}$ for a round: an I/O sum $S^{(i)}$ for a round is a modulo-two sum of a balanced binary-valued function f_i of the round input $Y^{(i-1)}$ and a balanced binary-valued function g_i of the round output $Y^{(i)}$, namely

$$S^{(i)} = f_i(Y^{(i-1)}) \oplus g_i(Y^{(i)})$$

Definition. *The imbalance $I(V)$ of a binary random variable V is the non-negative real number $|2P[V=0]-1|$ where $P[V=0]$ is the probability that V takes on the value 0.*

Definition HKM (Harpes-Kramer-Massey) [1]. The functions f_i and g_i are called input function and output function, respectively, of the I/O sum $S^{(i)}$. I/O sums for successive rounds will be called linked if the output function of each I/O sum except the last coincides with the input function of the following I/O sum: $g_i = f_{i+1}$. When $S^{(1)}, S^{(2)}, \ldots, S^{(r)}$ are linked, then their sum is also an I/O sum:

$$S^{(1\ldots r)} = \oplus_{i=1}^{r} S^{(i)} = f_0(Y^{(0)}) \oplus g_r(Y^{(r)})$$

which will be called an r-round I/O sum. Currently we observe these I/O sums and we apply the generalization of linear cryptanalysis presented in [1] to SAFER, and try to find effective homomorphic I/O sums for more than one round. SAFER is an r-round iterated cipher whose round function is defined in Figure.1.

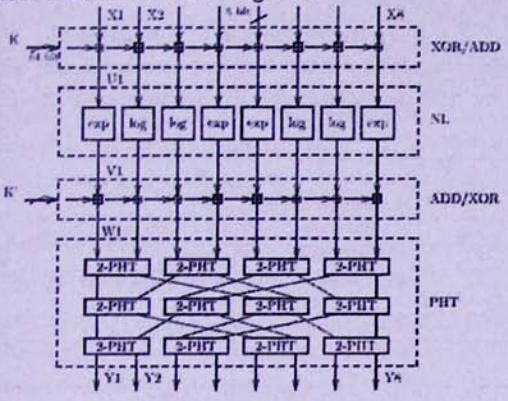

Figure 1: Round function of SAFER

Let X denote the input and Y the output of the round function. The round function consists of a cascade of the following operations:

1. a byte-wise mixed XOR/Byte-Addition (XOR/ADD) of 16 input bytes and 16 key bytes, the first part K of the round key – its output is $U = XOR/ADD(X,K)$
2. a non-linear layer, where each byte is subjected to either the non-linear function exp: $x \mapsto 45^x \bmod 257$ or its inverse function log – its output is $V = NL(U)$
3. a byte-wise mixed Byte-Addition/XOR (ADD/XOR) of 16 input bytes and 16 key bytes, the second part K' of the round key - its output is $W = ADD/XOR(V,K')$
4. a Pseudo-Hadamard Transform (PHT), consisting of four level "linear layer" boxes labeled "2-PHT", such that the output $Y = PHT(W)$

SAFER+ consists of the cascade of eight such rounds. In these notes, a binary-valued round-function f_i of the round input $Y^{(i-1)}$ can be divided into two "half-rounds", such that each half-round is the cascade of a keyed group operation (either XOR/ADD or ADD/XOR) and an unkeyed function (either Non-Linear or PHT). First, we have to find all homomorphisms for XOR/ADD or ADD/XOR respectively. There exist $2^{72}-1$ balanced homomorphisms for

ADD/XOR, namely the functions f_a defined by $f_a(V) = a \bullet V$, where a lies in the set of 128-bit tuples

$$A := \{a : a_1, a_4, a_5, a_8, a_9, a_{12}, a_{13}, a_{16} \in \{00,01\}; a_2, a_3, a_6, a_7, a_{10}, a_{11}, a_{14}, a_{15} \in F_{16}; a \neq 0\}$$

Similarly, there exist $2^{72} - 1$ balanced homomorphisms for XOR/ADD, namely the functions f_b where b lies in the set of 128-bit tuples

$$B := \{b : b_1, b_4, b_5, b_8, b_9, b_{12}, b_{13}, b_{16} \in F_{16}; b_2, b_3, b_6, b_7, b_{10}, b_{11}, b_{14}, b_{15} \in \{00,01\}; b \neq 0\}$$

We stress that the set of all homomorphic functions for XOR/ADD and the set of all homomorphic functions for ADD/XOR are subsets of the set of all linear Boolean functions. Therefore, the approach in the generalization of linear cryptanalysis is more restrictive than an attack analyzing all linear I/O sums. However, an analysis of all linear I/O sums is infeasible and we do not believe that it yields a better attack than ours. Secondly, we have to consider the half-round containing the PHT function. We denote these I/O sums as follows:

$$S_{a,b}^{PHT} := f_a(V) \oplus f_b(Y) \quad \text{where } a \in A \text{ and } b \in B$$

We define $S_{a,b}^{PHT}$ as guaranteed sum, if $I(S_{a,b}^{PHT}) = 1$, i.e. the imbalance of $S_{a,b}^{PHT}$ equals to 1. For the PHT-half-round, the only homomorphic I/O sums that have non-zero imbalances are $2^{16} - 1$ guaranteed I/O sums obtained by XOR-ing together any positive number of the 16 guaranteed I/O sums listed in Table 1.

Table 1: PHT Dependencies table

(a,b)	$f_b(Y)$	$f_a(W)$
(0111001100000000, 1000000000000000)	$Y1_0$	$W2_0 \oplus W3_0 \oplus W4_0 \oplus W7_0 \oplus W8_0$
(0111001111010000, 0100000000000000)	$Y2_0$	$W2_0 \oplus W3_0 \oplus W4_0 \oplus W7_0 \oplus W8_0 \oplus W9_0 \oplus W10_0 \oplus W12_0$
(1100000000000111, 0010000000000000)	$Y3_0$	$W1_0 \oplus W2_0 \oplus W14_0 \oplus W15_0 \oplus W16_0$
(1100110100000111, 0001000000000000)	$Y4_0$	$W1_0 \oplus W2_0 \oplus W5_0 \oplus W6_0 \oplus W8_0 \oplus W14_0 \oplus W15_0 \oplus W16_0$
(0000000011001101, 0000100000000000)	$Y5_0$	$W9_0 \oplus W10_0 \oplus W13_0 \oplus W14_0 \oplus W16_0$
(0011010011001101, 0000010000000000)	$Y6_0$	$W3_0 \oplus W4_0 \oplus W6_0 \oplus W9_0 \oplus W10_0 \oplus W13_0 \oplus W14_0 \oplus W16$
(0000000001110011, 0000001000000000)	$Y7_0$	$W10_0 \oplus W11_0 \oplus W12_0 \oplus W15_0 \oplus W16_0$
(1101000001110011, 0000000100000000)	$Y8_0$	$W1_0 \oplus W2_0 \oplus W4_0 \oplus W10_0 \oplus W11_0 \oplus W12_0 \oplus W15_0 \oplus W1$
(0000000011011100, 0000000010000000)	$Y9_0$	$W9_0 \oplus W10_0 \oplus W12_0 \oplus W13_0 \oplus W14_0$
(0100001111011100, 0000000001000000)	$Y10_0$	$W2_0 \oplus W7_0 \oplus W8_0 \oplus W9_0 \oplus W10_0 \oplus W12_0 \oplus W13_0 \oplus W14$
(0000110100110000, 0000000000100000)	$Y11_0$	$W5_0 \oplus W6_0 \oplus W8_0 \oplus W11_0 \oplus W12_0$
(0000110100110111, 0000000000010000)	$Y12_0$	$W5_0 \oplus W6_0 \oplus W8_0 \oplus W11_0 \oplus W12_0 \oplus W14_0 \oplus W15_0 \oplus W16$
(1101110000000000, 0000000000001000)	$Y13_0$	$W1_0 \oplus W2_0 \oplus W4_0 \oplus W5_0 \oplus W6_0$
(1101110000001101, 0000000000000100)	$Y14_0$	$W1_0 \oplus W2_0 \oplus W4_0 \oplus W5_0 \oplus W6_0 \oplus W13_0 \oplus W14_0 \oplus W16_0$
(0011011100000000, 0000000000000010)	$Y15_0$	$W3_0 \oplus W4_0 \oplus W6_0 \oplus W7_0 \oplus W8_0$
(0011011101110000, 0000000000000001)	$Y16_0$	$W3_0 \oplus W4_0 \oplus W6_0 \oplus W7_0 \oplus W8_0 \oplus W10_0 \oplus W11_0 \oplus W12_0$

Secondly, $S_{a,b}^{PHT}$ depends linearly on some W_α if $f_b(PHT(W))$ depends linearly on this W_α and if $a_\alpha = 0$. Within our research we created the dependencies table between certain bits of the PHT output vector on certain bits of the PHT input vector.

Let now show, that besides of these $2^{16}-1$ guaranteed I/O sums all other sums have zero imbalance. The random variable $f_b(Y)$ can be written as a XOR-sum of the output bits $Y1_7, Y1_6, ..., Y16_0$ and $f_a(W)$ as a XOR-sum of the input bits $W1_7, W1_6, ..., W16_0$. All 72 output bits that may appear in the sum for $f_b(Y)$ if $b \in B$ can be written as the binary-valued expression containing input bits, but no output bits. In particular, we are interested in those 56 input bits that can never appear in $f_a(W)$ if $a \in A$. In Table 2, we consider a column for each of these input bits and a row for each of the former output bits. The entry of the row Y_β and the column W_α is:

0 if we know that Y_β is independent of W_α, i.e. if the expression for Y_β does not contain W_α;

1 if Y_β is linear in W_α, i.e. if we can write $Y_\beta = W_\alpha \oplus \phi(W_0, ..., W_{\alpha-1}, W_{\alpha+1}, ... W_{128})$;

? otherwise

Let $S^{PHT}_{a,b} := f_a(W) \oplus f_b(Y)$ have non-zero imbalance. Consider the column for $W0_7$. It contains only zeros but three "1" in rows for $Y3_7, Y7_7$ and $Y12_7$. There is

$$f_b(Y) = b \bullet Y7 = b3_7 Y3_7 \oplus b_7 Y7_7 \oplus b12_7 Y12_7$$

relation, where $f_b(Y)$ is linearly dependent when $Y3_7 \oplus Y7_7 \oplus Y12_7 = 1$.

It is clear, that for each values of $b3_7, b7_7, b12_7 \in \{0,1\}$ where $(b3_7, b7_7, b12_7) \neq (0,0,0)$, $f_b(Y)$ function has zero imbalance. As $f_a(W)$ is independent of $W0_7$, $S^{PHT}_{a,b}$ is again linear in $W0_7$ and its imbalance is zero. So, by contradiction, $(b3_7, b7_7, b12_7) = (0,0,0)$ and we can ignore the rows $Y3_7, Y7_7$ and $Y12_7$ in our further analysis. The iterative analysis shows that there is no non-balanced homomorphic I/O sum in this regards.

We next consider the half-round containing the non-linear function NL and tried to find homomorphic I/O sums for NL that have non-zero imbalance. Such I/O sums can be obtained by summing I/O sums for its building blocks EXP and LOG. For the function EXP with input U_1 and V_1 output, the only homomorphic I/O sums are

$$S^{EXP}_{a_1,b_1} = (a_1 \bullet U_1) \oplus (b_1 \bullet V_1), \quad \text{for } a_1 \in B^{16} \setminus \{00\} \text{ and } b_1 = 01.$$

The most effective ones are obtained when (a_1, b_1) is equal to (cd,01) or (ff,01) (the imbalance being 28/128) or to (86,01), (bf,01), (c0,01) or (f7,01) (the imbalance being 24/128).

Remark. $I(S^{EXP}_{01,01}) = I(S^{EXP}_{02,01}) = I(S^{EXP}_{03,01}) = 0$. Furthermore, for all $a_1, b_1 \in B^{16}$, if $a_1[7] = 0$, then $I(S^{EXP}_{a_1,b_1}) = 0$.

For the function LOG with input U_2 and output V_2, the only homomorphic I/O sums are

$$S^{LOG}_{a_2,b_2} = (a_2 \bullet U_2) \oplus (b_2 \bullet V_2) \quad \text{for } a_2 = 01 \text{ and } b_2 \in B^{16} \setminus \{00\}.$$

Their imbalances are easily deduced since $I(S^{EXP}_{a_1,b_1}) = I(S^{LOG}_{b_1,a_1})$. Finally, we have to link I/O sums for successive half-rounds.

The following theorem takes place (see Theorem1 in [2]).

Theorem 1. *The procedure for finding effective homomorphic I/O sums does not find an I/O sum with non-zero imbalance for a cascade of half-rounds taken in the same order as they are used in SAFER and containing at least two PHT-layers.*

Task paralleling in the program is realized in the most optimal way as possible. One of the processes is considered to be the main: it distributes permutations between processes and registers the output of cryptanalysis. Upon receiving the permutations sent by the main process, the other processes perform their linear cryptanalysis and inform the main process on the obtained results. The program provides possibility to obtain new "Armenian Shuffle" coordinate permutations which ensure good diffusion and security of cryptosystems against linear cryptanalysis. As the obtained "Armenian Shuffle" coordinate permutations where previously chosen as secure against differential cryptanalysis, so the cryptosystems constructed with these permutations are secure and can be used for data coding.

References:

1. J. L. Massey, G. H. Khachatrian and M. K. Kuregian, ``Nomination of SAFER+ as Candidate algorithm for the Advanced Encryption Standard (AES)", Submission document from Cylink Corporation to NIST, June 1998.
2. J. L. Massey, G. H. Khachatrian and M. K. Kuregian, ``Nomination of SAFER++ as Candidate Algorithm for the New European Schemes for Signatures, Integrity, and Encryption (NESSIE)", Submission document from Cylink Corporation, 2000.
3. C. Harpes, ``Cryptanalysis of iterated block ciphers", ETH Series in Information Processing, editor: James L. Massey. v. 7, Hartung-Gorre Verlang Konstanz, 1996.
4. C. Harpes, G. G. Kramer and J. L. Massey, ``A generalization of linear cryptanalysis and the applicability of Matsui's piling-up lemma", Presented at Eurocrypt '95.
5. C. Harpes, ``A generalization of linear cryptanalysis applied to SAFER", Signal and Info. Proc. Lab., CH-8092 Zurich, March 9, 1995.

Բլոկային ծածկագրման համակարգերի գծային վերծանումը կլաստերային հաշվողական համակարգում

Մ. Կյուրեղյան, Օ. Մանուկյան և Է. Հարությունյան

Ամփոփում

Աշխատանքում նկարագրված են արդյունքներ SAFER+ և SAFER++ բլոկային ծածկագրման համակարգերի նոր տարբերակի կառուցման վերաբերյալ: Ստեղծվել է զուգահեռ հաշվարկների "LinearCryptanalyser" ծրագրաշար, որի օգնությամբ փնտրվում են այնպիսի "Armenian Shuffle" կոորդինատային տեղափոխություններ, որոնց համապատասխան բլոկային ծածկագրման համակարգերը կլինեն կայուն դիֆերենցիալ և գծային վերլուծությունների նկատմամբ: Հետազոտությունները կատարվել են կապակցված մուտք/ելք գումարների գոյության ինչպես տեսական, այնպես էլ կիրառական տեսանկյուններից: "LinearCryptanalyser" զուգահեռ հաշվարկների ծրագրաշարը թույլ է տալիս ուսումնասիրել կապակցված մուտք/ելք գումարների գոյության հարցը, իսկ նման կապակցված գումարների բացակայությունը վկայում է ուսումնասիրվող բլոկային ծածկագրման համակարգերի կայունության մասին գծային կրիպտոանալիզի նկատմամբ: