Proceedings of ACS'2000, Szczecin, pp.523-530

NEW CONSTRUCTIONS IN LINEAR CRYPTANALYSIS OF BLOCK CIPHERS[‡]

ANNA ZUGAJ[1], KAROL GÓRSKI[1], ZBIGNIEW KOTULSKI[2], ANDRZEJ PASZKIEWICZ[3], JANUSZ SZCZEPAŃSKI[2]

[1]ENIGMA Information Security Systems Sp. z o.o., Cryptography Dept., ul. Orla 11/15, 00-143 Warsaw, POLAND, ph./fax: (+48 22) 863 62 65, email: {ania, karol}@enigma.com.pl
[2]Institute of Fundamental Technological Research, Polish Academy of Sciences, ul. Świętokrzyska 21, 00-049 Warsaw, POLAND, ph.: (+48 22) 826 12 81, fax: (+48 22) 826 98 15, email: {zkotulsk, jszczepa}@ippt.gov.pl
[3]Institute of Telecommunications, Warsaw University of Technology, ul. Nowowiejska 15/19, 00-665 Warsaw, POLAND, ph.: (+48 22) 660 78 35, fax: (+48 22) 825 49 50, email: anpa@tele.pw.edu.pl

Abstract. At the beginning of the paper we describe the state of art in linear cryptanalysis of block ciphers. We present algorithms for finding best linear expressions proposed by Matsui [9] and Ohta [11]. We sketch basic linear cryptanalysis (0R, 1R, 2R attacks) and the known extensions. We explain the advantages and the limitations of applying linear cryptanalysis and its extensions to block ciphers.
In the second part of the paper we describe our proposal of a new extension to linear attack based on the application of a probabilistic counting method. It allows the reduction of two consecutive rounds and form the basis for mounting e.g. 3R attacks. We present experimental results of the implementation of this attack to the Data Encryption Standard.

Keywords: block cipher, linear cryptanalysis, linear expression, probabilistic counting method, Data Encryption Standard

1. INTRODUCTION

Symmetric block ciphers are one of the fundamental tools in modern cryptography. Their popularity requires a high level of trust in their security. Unfortunately there are neither any known constructions of block ciphers, which offer unconditional security nor practical constructions, which offer provable computational security. So in practice evaluations of the security of these ciphers is heuristic based on the consideration of the resistance of the cipher to known attacks. The effectiveness of attacks is measured by comparison of their complexity (time and memory) with the exhaustive search attack. During this evaluation only those attacks are taken into account, which are known at the time. One of the most important attacks considered is linear cryptanalysis. In 1993 it was successfully used by Matsui to cryptanalyse DES. It needed 2^{43} known plaintext/ciphertext pairs to derive 26 bits of the key.
The purpose of this paper is to describe the main issues of linear cryptanalysis beginning from algorithms for finding best linear expressions [9,11] through description of basic attack (0R) and linear attacks with round reductions (1R, 2R) to various extensions of linear cryptanalysis (analysis with multiple expressions [5], linear-differential cryptanalysis [7], linear cryptanalysis with non-linear approximations in outer rounds [6], the use of quadratic relations in S-box [13] and the probabilistic counting method [12]) and the limitations of their use.

We also propose the application of probabilistic counting method for reduction of two consecutive rounds which forms the basis for mounting e.g. 3R attacks (which in some applications are more effective – require less texts than 2R attacks - even though they are based on probabilistic assumptions). We describe the implementation of this attack to Data Encryption Standard. The results should be treated as announcements only, because the experiments are still under development.

1.1 Notation and Definitions

Throughout this paper we use Matsui's [8] numbering of DES bits. The input bits, key bits and output bits of F-functions, S-boxes, etc. are

[‡] This work has been supported by grant No 8 T11D 020 19 of the Polish Scientific Research Committee.

numbered from right to left starting from 0. We also use Matsui's notation in which $A[i]$ denotes i-th bit of vector A, while $A[i_1, i_2, ... i_n]$ denotes exclusive-or of the bits of vector A located in positions $i_1, i_2, ... i_n$. We also use the notation of Harpes [4] in which $A \bullet \Gamma A$ denotes scalar multiplication of two binary vectors over GF(2), which is equivalent to exclusive-or of the A bits chosen by binary vector ΓA (e.g. A = 1011, ΓA = 0001, then $A \bullet \Gamma A = 0 \oplus 0 \oplus 0 \oplus 1 = A[i_4]$).

Let P, C, K denote plaintext, ciphertext and key. We assume that plaintexts, ciphertexts and keys are uniformly distributed in appropriate spaces. We also assume that round keys are independent.

By r we denote the number of rounds, while by C_i we denote the ciphertext after round i, which means that $P = C_0$ and $C = C_r$. N denotes the number of analysed pairs of texts.

A linear approximation is a linear dependence between bits of the round input block, bits of the round output block and bits of the round subkey. A linear expression is a linear dependence between bits of the cipher input, cipher output and bits of all the subkeys. An effective linear expression is an expression which holds with probability different from 1/2.

Probability of the linear approximation (p) is defined in the probabilistic space with:

- a set of elementary events Ω, which is a Cartesian product of the set of all input blocks to the round and all subkey blocks,
- σ - field which is the set of all subsets of Ω,
- probability distribution on the elementary events assigning to each of them equal probability.

There is a random variable defined in this space, which assigns to each elementary event the value 0 or 1, dependent on whether the approximation holds or not. Event X is defined as a sum of the elementary events for which the random variable is equal to 1. Probability of a linear approximation is equal to the probability of event X in this probabilistic space.

1.2 Linear Cryptanalysis

The basic idea of linear cryptanalysis is to find an effective linear expression for an analysed block cipher, s.t.:

$$(P \bullet \Gamma P) \oplus (C \bullet \Gamma C) = \Sigma_z (K_z \bullet \Gamma K_z). \qquad (1)$$

with a certain probability p, measured over all choices of P and K.

In the case of iterative block ciphers, finding the linear expression has 2 steps. At first we linearise one round, looking for effective approximations of the following form:

$$(C_{i-1} \bullet \Gamma C_{i-1}) \oplus (C_i \bullet \Gamma C_i) = K_i \bullet \Gamma K_i \qquad (2)$$

where C_{i-1} is the input vector to round i, C_i is the output vector from round i and K_i is the key used in round i. A linear expression is obtained by combining linear approximations in such a way that only bits of plaintext, ciphertext and subkeys appear in the final expression. For a few rounds of a cipher and for ciphers with a simple structure (e.g. RC5) this process can be done manually, but in most cases it is easier to use a computer. The algorithms for finding linear expressions for DES [3] are described below.

With an effective linear expression we can start a so-called 0R attack (algorithm 1), based on the maximum likelihood method. This attack determines with required probability whether the right side of equation 1 is equal to 0 or 1. The success rate of the attack increases with the number of analysed texts and with the bias $|p - 1/2|$.

Algorithm 1 (attack 0R) [8]

Input:

N known pairs of plaintext and ciphertext,

effective linear expression with probability p

Step 1:

For each pair count the value of left side of equation 1. Let N_0 be the number of pairs for which the left side of the equation is equal to 0.

Step 2:

If $N_0 > N/2$ then

set $\Sigma_i(K_i \bullet \Gamma K_i) = 0$, if $p>1/2$ and 1 if $p<1/2$,

else

set $\Sigma_i(K_i \bullet \Gamma K_i) = 1$, if $p>1/2$ and 0 if $p<1/2$.

Output:

the value of $\Sigma_i(K_i \bullet \Gamma K_i)$ (correct with probability dependent on N and $|p - 1/2|$).

In practical attacks with similar complexity we can obtain more subkey bits. For this purpose attacks with round reduction are used (1R and 2R). The first uses an effective linear expression for r-1 rounds and computes the inverse of the last round of the cipher for each candidate for the last round subkey. For each candidate we count the difference between the number of times when the left side of the linear expression is equal to 0 and when it is equal to 1. For the correct subkey the bias between this value and N/2 will be close to the expected bias for the expression in use. For incorrect keys it will be close to 0. In this way we can determine with the required probability the subkey bits in the last round and the value of the modulo 2 sum of the subkey bits appearing in the linear expression. The idea of this attack is based on a hypothesis described by Harpes [4] that the choice of an incorrect key in the last round is equivalent to adding an additional round to the cipher, which decreases the effectiveness of the linear expression in use. In practice checking all the possible values of the subkey in the last round is too complex (requires too much memory). The solution is to check only a subset of the bits of the last round subkey.

In a similar way the 1R attack can be used for the reduction of the first round of the cipher.

Algorithm 2 (attack 1R) [8]

Input:

N known pairs of plaintext and ciphertext,

effective subset of last round subkey bits being searched, effective linear expression for r-1 rounds with probability p, which uses only these bits of C_{r-1} which can be computed from the effective subset of subkey bits

Step 1:
For value of K^i_r effective bits of subkey K_r, let N_0^i denote the number of pairs of texts for which the left side of the (r-1) - round linear expression is equal to 0.

Step 2:
Let $N_{0max} = \max_i (N_0^i)$ and $N_{0min} = \min_i (N_0^i)$.

Step 3:
If $|N_{0max} - N/2| > | N_{0min} - N/2 |$ then
set the value of effective subkey bits K^i_r corresponding to N_{0max},
set $\Sigma_i(K_i \bullet \Gamma K_i) = 0$, if $p>1/2$ and 1 if $p<1/2$,
If $|N_{0max} - N/2| < | N_{0min} - N/2 |$ then
set the value of effective subkey bits K^i_r corresponding to N_{0min},
set $\Sigma_i(K_i \bullet \Gamma K_i) = 1$, if $p>1/2$ and 0 if $p<1/2$,

Output:
effective subkey bits in last round,
the value of $\Sigma_i(K_i \bullet \Gamma K_i)$ for rounds 1 to r-1,
both results returned with probability dependent on N and $|p-1/2|$.

The 2R attack allows further increase of the effectiveness of the analysis. The idea is similar to the 1R attack: we use an expression for r-2 rounds of the cipher and invert the first and the last round.

To give a sketch of probabilistic fundamentals we recall here the Piling–Up Lemma, which is used to calculate probability p of the linear expression, when the probabilities p_i $(1 \leq i \leq r)$ of all linear round approximations are known:

Lemma 1 (Piling-Up) [8]
Let $Appr_i$ $(1 \leq i \leq r)$ be independent, random variables, which are equal to 0 with probability p_i and are equal to 1 with probability $1 - p_i$. Then the probability that

$$Appr_1 \oplus Appr_2 \oplus ... \oplus Appr_r = 0 \quad (3)$$

is equal to:

$$1/2 + 2^{r-1} \prod_{i=1}^{r} (p_i - 1/2). \ \square \quad (4)$$

Then the probability of proper choice of key bits xor in 0R attack is equal to:

$$\Pr(N_0 > N/2) = \frac{1}{\sqrt{2\pi}} \int_{-2\sqrt{N}(p-1/2)}^{\infty} e^{-t^2/2} dt. \quad (5)$$

This equation describes the success rate (Table 1) for some probability p of a linear expression. This probability increases when the number of analysed texts increases and when bias $|p-1/2|$ increases.

Table 1. Success rate of 0R attack

N	$¼\|p-½\|^{-2}$	$½\|p-½\|^{-2}$	$\|p-½\|^{-2}$	$2\|p-½\|^{-2}$
probability of success	84,1%	92,1%	97,7%	99,8%

In linear cryptanalysis with 1 round reduction the probability of the correct choice of subkey bits is equal to:

$$\Pr(K^i_{r\,max} = k_r) = \frac{1}{\sqrt{2\pi}} \int_{-2\sqrt{N}(p-1/2)}^{\infty} \Big(\prod_{K_r^i \neq k_r} \int_{-x-4\sqrt{N}(p-1/2)q^i}^{x+4\sqrt{N}(p-1/2)(1-q^i)} \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy \Big) e^{-x^2/2} dx$$

The above equation describes the success rate (Table 2) of the 1R attack.

Table 2. Success rate of 1R attack

N	$2\|p-½\|^{-2}$	$4\|p-½\|^{-2}$	$8\|p-½\|^{-2}$	$16\|p-½\|^{-2}$
probability of success	48,6%	78,5%	96,7%	99,9%

For further details of the probabilistic fundamentals of linear cryptanalysis see [17].

2. EXPRESSION SEARCH ALGORITHM

As we mentioned above the first step in a linear attack is to find an effective linear expression for the cipher. This should be done by linearising the non-linear elements and extending this linearisation to the beginning and end of the round function. Now the propagation of the masking values should be considered. We give an example on DES:

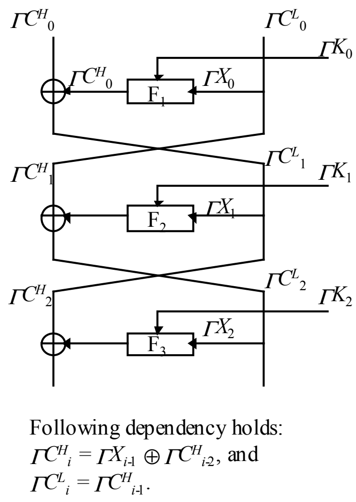

Fig. 1. Propagation of masking values in DES.

It is very important to notice that propagation of masking values (Fig. 1) is different from the operations in the cipher, e.g. when we consider a mask as a set of bits, we will see that an exclusive-or operation on a mask will work as a tee, while a tee operation will work like an exclusive-or.

At the beginning of expression search algorithm we set a boundary value of the expression bias. We also set a value of the best one round bias. During the analysis, we will compare the bias of the current expression concatenated with the best possible expression with the boundary bias. If our current expression would not be better that the bound, we will discard it.

In the first round we choose ΓC^H_0 in such a way as to determine the approximation of the round with the largest bias $|p - 1/2|$. In other words we choose ΓC^H_0, and we try to find such ΓX_0 that the bias is large. If the chosen approximation concatenated to the best r-1 round expression would have better bias than the bound we can start looking for the approximation of the second round. Otherwise we have to try find a better approximation for the first round.

In the second round we have a similar situation, we control the masking value ΓC^H_1 (through masking value ΓC^L_0) in such a way as to find the approximation of the second round which concatenated to the expression of (r-2) rounds would give better bias than the bound. In the following rounds we have ΓC^H_i fixed, we can only choose ΓX_i to improve the bias.

At the end of the algorithm we get one or more linear expressions and we can start analysis.

The algorithm sketched above was presented by Matsui [9]. Ohta [11] optimised this algorithm by discarding some expressions during precomputation phase. He obtained a significant improvement in the expression search of FEAL. The comparison of the effectiveness of these algorithms for searching for linear expressions for DES can be found in [14].

3. EXTENSIONS OF LINEAR CRYPTANALYSIS

Several extensions to linear cryptanalysis were proposed, which improve the effectiveness of the attack, e.g. use of non-linear approximations in outer rounds reduces the number of analysed texts by a factor of $1/\sqrt{2}$.

Differential-linear cryptanalysis is a very powerful attack on DES with a reduced number of rounds. The uses only 512 chosen plaintexts in comparison to linear cryptanalysis which needs to analyse 500,000 of known plaintexts and to differential cryptanalysis which needs to analyse 5,000 chosen plaintexts to obtain the same success probability.

Multiple expression[1] attack reduces the number of analysed texts by a factor of $\dfrac{p-1/2}{\sqrt{\sum_i (p_i - 1/2)^2}}$, where p is the probability of the best linear expression in use, and p_i are the probabilities of each of the expressions.

The latest extension proposed by Shimoyama [13] reduces the number of plaintexts by the factor 25/34.

In this section we sketch all these attacks.

[1] called multiple approximation in [5].

3.1 Non-linear approximations in outer rounds

It was natural to consider whether linear approximations in linear cryptanalysis can be replaced by non-linear ones. There are two advantages which this extension could give. Firstly, the number of non-linear approximations is much larger than linear ones, so it may be easier to find an approximation with a large bias. Secondly, it would make possible an attack on large S-boxes used in round functions. Unfortunately, Harpes [4] demonstrated problems in general use of non-linear approximations in linear analysis.

Knudsen proposed to use non-linear approximations in outer rounds. He used only approximations with a non-linear combination of the input bits and a linear combination of the output bits.

For an illustration of the attack we present an example. Consider an approximation of DES S-box S_8 which involves bits x_0x_1 on the S-box input. Then, depending on the value of the subkey bits k_0 and k_1 and denoting the appropriate text bits after expansion by z_0 and z_1, we obtain that $x_0x_1 = z_0z_1$, when $(k_0, k_1) = (0, 0)$, $x_0x_1 = z_0z_1 \oplus z_0$, when $(k_0, k_1) = (1, 0)$, $x_0x_1 = z_0z_1 \oplus z_1$, when $(k_0, k_1) = (0, 0)$ and $x_0x_1 = z_0z_1 \oplus z_0 \oplus z_1 \oplus 1$, when $(k_0, k_1) = (1, 1)$.

In outer rounds the cryptanalyst knows the value corresponding to bits z_0, z_1 before the transformation with the subkey. Similarly to the 1R attack, he can try to guess the value of the correct subkey bits. Assume that the probability of the approximation in use is equal to p. When his guess is correct, he correctly reconstructs x_0 and x_1 and the product x_0x_1. When his guess is incorrect, e.g. he chooses $k_0 \oplus 1$ and k_1, then he guesses $(x_0 \oplus 1)x_1$ and the expression on input bits to the S-box will be equal to the expression on output bits of the S-box with some probability p_1. If $|p_1\text{-}1/2| < |p\text{-}1/2|$ then with a sufficient number of analysed texts the incorrect choice can be detected. In the opposite case the incorrect guess will dominate, but in a practical attack the cryptanalyst chooses the approximation with a larger bias anyway. When both biases are equal, they are indistinguishable for the cryptanalyst.

Knudsen applied his attack to DES reduced to five rounds; the comparison with the original attack is given in the following tables:

Table 3. Success rate in 0R Matsui's attack on 5 rounds of DES ($(p\text{-}½)^{-2}$ = 68,720)

N	17,180	34,360	68,720
probability of success	74%	88%	98%

Table 4. Success rate in 0R attack on 5 rounds of DES with non-linear approximation in outer rounds ($(p\text{-}½)^{-2}$ = 14,728)

N	3,682	7,364	14,728
probability of success	86%	92%	100%

3.2 Differential-linear cryptanalysis

Differential-linear cryptanalysis was proposed by Langford and Hellman [7]. They noticed that three round differential characteristics [1] which hold with probability 1 can be effectively used in linear cryptanalysis.

The main idea of the attack is the observation that complementing two bits (which after expansion are the middle bits of an input to the S-box) in one of the analysed texts leaves many bits of C_3 unchanged. Among these bits are input bits to Matsui's best 3-round linear expression (bits number 57, 46, 40, 35 and 17). Because the parity of these bits never changes, the parity of output bits from the linear expression is unchanged with probability $p' = p^2 \oplus (1-p)^2 = 0.576$, where $p = 0.695$ is the probability of Matsui's linear expression. (This result comes directly from the Piling-Up Lemma.)

To attack DES the cryptanalyst for each pair of ciphertexts inverts the last round, computes the parity for both inverted ciphertexts and, if the parity is equal increases N_0^j where j is the index of the analysed candidate for the last round subkey. The largest N_0^j indicates the correct subkey with a probability depending on the probability of the linear expression in use and the number of analysed pairs.

Further improvement of this attack can be achieved by using structures proposed by [1] for packing the analysed plaintexts. To sketch the idea of structures we give an example. When there is a possibility to use more than one differential characteristic in an attack e.g. 4-tuples of plaintexts: P, P ⊕ 0x2000000000000000, P ⊕ 0x4000000000000000, P ⊕ 0x6000000000000000, instead of time-consuming encryption of all these plaintexts, we can encipher only three of them and get the information about the 4-th through analysis, which is not so time-consuming.

3.3 Multiple expressions

The extension proposed by Kaliski and Robshaw [5] was based on the observation that during the attack, the cryptanalyst differentiates between the distribution with an expected value equal to p and variance p^2 and the distribution with an expected value equal to 1-p and variance p^2. Use of multiple expressions decreases the variance of the distributions.

Modified equation 1 assumes the following form:

$$(P \bullet \Gamma P^j) \oplus (C \bullet \Gamma C^j) = \Sigma_i (K_i \bullet \Gamma K_i), \qquad (6)$$

where ΓP^j, ΓC^j denote binary masking vectors of plaintext and ciphertext used in linear expression number j ($1 \le j \le J$).

Instead of N_0 in algorithm 1, Kaliski proposed to use a statistic of the following form:

$$U = \sum_{j=1}^{J} a_j N_0^j \qquad (7)$$

where $a_1, a_2, ..., a_J$, are positive and s.t. $\sum_{j=1}^{J} a_j = 1$.

For simplicity we assume that p_j-1/2 > 0.

Algorithm 3 (attack 0R with multiple expressions) [5]

Input:
  N known pairs of texts,
  effective linear expressions with probability p_j.

Step 1:
  For each linear expression let N_0^j be the number of pairs for which the left side of equation 6 was equal to 0.

Step 2:
  Count the value $U = \sum_{j=1}^{J} a_j N_0^j$.

Step 3:
  If $U > N/2$ then
    set $\Sigma_i(K_i \bullet \Gamma K_i) = 0$, if $p>1/2$ and 1 if $p<1/2$,
  else
    set $\Sigma_i(K_i \bullet \Gamma K_i) = 1$, if $p>1/2$ and 0 if $p<1/2$.

Output:
  the value of $\Sigma_i(K_i \bullet \Gamma K_i)$ (correct with probability dependent on N and $|p - 1/2|$ and weights a_j.

Kaliski noticed that the distribution of statistic U can be modelled using a normal distribution. He calculated the expected values and the variance. He also indicated that when the weights a_j are proportional to the biases (p_j-1/2) of linear expressions, the distance between N/2 and E[U] is maximised. He calculated the success rate of the modified algorithm, which is equal to:

$$\Phi\left(2\sqrt{N}\sqrt{\frac{\sum_{j=1}^{n}(p_j-1/2)^2}{1-4\sum_{j=1}^{n}(p_j-1/2)^2}}\right), \qquad (8)$$

where $\Phi(.)$ denotes the normal cumulative distribution function. When $\sum_{j=1}^{J}(p_j - 1/2)$ is small, the success rate can be approximated as $\Phi(2\sqrt{N}\sqrt{\sum_{j=1}^{n}(p_j-1/2)^2})$, while the success rate of Matsui's algorithm is equal to $\Phi(2\sqrt{N}(p-1/2))$.

Algorithm 3 can be easily extended to 1R and 2R attacks.

3.4 Shimoyama's attack

Recently Shimoyama [13] proposed an extension using formal coding of DES S-boxes to invert an outer round with probability 1. He found that there are seven algebraic quadratic relations of the DES S-boxes. He used one of these relations instead of the outer approximation in a linear expression. His approximation which was used to invert S-box S5 has the following form:

$(y_1 \oplus y_2 \oplus y_3 \oplus y_4 \oplus x_2 \oplus 1) * (x_1 \oplus x_2 \oplus x_5 \oplus 1) = 0$

which gives on the input and output to F function:

$(F_i[3,8,15,24,] \oplus C_i[17] \oplus K_i[26] \oplus 1)$
$*(C_i[16,17,20] \oplus K_i[25,26,29] \oplus 1) = 0.$

Shimoyama used this relation instead of first round approximation in a linear expression of DES. He estimated each of the factors independently, which reduced the memory requirements. And finally he combined the results of both factors using Kaliski's [5] method.

3.5 Limitations of the basic attack and its extensions

The basic attack and its extension have the following limitations:

1. Complexity of the non-linear approximation search algorithms – effective search is feasible only then, when non-linear operations are algebraically defined e.g. as an addition in some field, or when the number of all possible combination of input bits to the operation is small.
2. Memory complexity of the attack – it is usually impossible to mount an attack with two consecutive round reduction (due to mixing property e.g. in DES due to construction of permutation P). In attack on DES the cryptoanalyst needs to implement $2^{6*6+6}=2^{42}$ counters for candidates for a subkey in two consecutive rounds.
3. Computational complexity $\boldsymbol{O}(N)$.

To make an attack more flexible the cryptoanalyst needs to achieve an independence between the number of effective subkey bits and the multiple of the number of inputs to the S-box. It can be realised by use of non-linear approximations (but we have to remember the limitation due to complexity of non-linear approximation search algorithms) or by use of probabilistic counting method [12], which is described in the following chapter.

4. NON-DETERMINISTIC APPROACH

We describe linear cryptanalysis with probabilistic counting method applied to DES. Use of this method increased the flexibility in the choice of number of effective key bits (in reduced rounds). We propose the construction of the attack with the reduction of two consecutive rounds. Such a construction can be effective due to use of the probabilistic counting method for an inversion of inner (second or penultimate) round of the cipher. This attack with the reduction of two consecutive rounds form the basis for mounting 3R attack. It makes possible to use the linear expression for the smaller number of rounds e.g. for (r-3) rounds, and for the reduction of number of analysed pairs of texts.

4.1 Linear cryptanalysis with the probabilistic counting method

In linear cryptanalysis with the probabilistic counting method it is presumed that analyst knows only a part of effective key bits (which are called visible bits)[2]. Bits unknown for the analyst are called invisible bits. During estimation of the value of the expression in which invisible effective key bits appear, instead of exact value its approximation is used, which holds with some probability.

To explain the probabilistic counting method we present an example with 1R attack. We use (r-1)-round effective linear expression and the approximation of one S-box in last or first round. Similarly to the previously discussed attacks (basic linear cryptanalysis and its extensions) for each part of subkey (visible effective subkey bits) we determine the bias between the number of events (pairs of texts) in which left side of the (r-1)-round linear expression is equal to 0 and the number of events when it is equal to 1. For proper subkey in outer round the bias should be close to the bias expected for the expression and for the wrong keys it should be close to 0. In this way we can conclude with required probability about the subkey bits in outer round and about the value of the exclusive-or of subkey bits in remaining (r-1) rounds.

Let an (r-1)-round effective linear expression satisfied with probability $p \neq 0{,}5$, have the following form:

$$P \bullet \Gamma P \oplus C_{r\text{-}1} \bullet \Gamma C_{r\text{-}1} = \Sigma_i\, K_i \bullet \Gamma K_i, \qquad (9)$$

from which we can obtain:

$$P \bullet \Gamma P \oplus C^L_r \bullet \Gamma C^H_{r\text{-}1} \oplus (\mathrm{F}(C^L_r, K_r) \oplus C^H_r) \bullet \Gamma C^L_{r\text{-}1} = \Sigma_i\, K_i \bullet \Gamma K_i. \qquad (10)$$

In attack with probabilistic counting method instead of exact value $\mathrm{F}(C^L_r, K_r) \bullet \Gamma C^L_{r\text{-}1}$ we use its approximation. We denote this approximation as $\sim(\mathrm{F}(C^L_r, K_r) \bullet \Gamma C^L_{r\text{-}1})$, and finally for an attack with probabilistic counting we obtain the expression:

$$P \bullet \Gamma P \oplus C^L_r \bullet \Gamma C^H_{r\text{-}1} \oplus C^H_r \bullet \Gamma C^L_{r\text{-}1} \oplus \sim(\mathrm{F}(C^L_r, K_r) \bullet \Gamma C^L_{r\text{-}1}) = \Sigma_i\, K_i \bullet \Gamma K_i. \qquad (11)$$

The probability of the expression (11) depends on probability p of (r-1)-rounds linear expression and probability of probabilistic approximation of F function.

Let K^v_r denote the key candidate for effective visible subkey bits in round r. Then the algorithm of linear cryptanalysis with probabilistic counting for DES has the following form:

Algorithm 4 (1R attack with probabilistic counting and data counting phase)

<u>Data counting phase</u>

Input:

N known pairs of texts,

effective linear expression for (r-1)-rounds with probability p, which uses only these bits of $C_{r\text{-}1}$,

[2] In practice this situation can occur, when an analyst has limited memory resources.

which can be computed from subset of effective bits k_r^v.

Step 1:

Prepare 2^t counters T_j ($0 \le j < 2^t$) and initiate them with zero, where j denotes value of effective text bits used in linear expression

Step 2:

For each plaintext and suitable ciphertext count t and increment value of counter T_j.

Output:

Counter table T_j.

Key counting phase

Input:

table T_j,

choice of effective subkey bits K_r,

choice of visible effective subkey bits K_r^v, which are being searched,

effective linear expression for (r-1) rounds with probability p which uses only these bits of C_{r-1}, which can be computed from subset of effective bits k_r^v.

Step 3:

Prepare 2^{k_v} counters N_0^i ($0 \le i < 2^{k_v} - 1$) and initiate them with 0.

Step 4:

For each possible value i of effective subkey bits K_r^v and for each possible value j of effective text bits count the probability p_{ij} that left side of the linear expression assumes value zero averaged over all invisible effective key bits. Then set counter $N_0^i = \Sigma_j\, p_{ij} * T_j$.

Step 5:

Set $N_{0max} = \max_i (N_0^i)$ and $N_{0min} = \min_i (N_0^i)$.

Step 6:

If $|N_{0max} - N/2| > |N_{0min} - N/2|$ then

set the value of effective visible subkey bits K_r^i corresponding to N_{0max},

set $\Sigma_z (K_z \bullet \Gamma K_z) = 0$, if $p>½$ or 1, if $p<½$.

If $|N_{0max} - N/2| < |N_{0min} - N/2|$ then

set the value of effective visible subkey bits K_r^i corresponding to N_{0min},

set $\Sigma_z (K_z \bullet \Gamma K_z) = 1$, if $p>½$ or 0, if $p<½$.

Output:

effective visible subkey bits in last round,

the value of ($\Sigma_z (K_z \bullet \Gamma K_z)$) for rounds 1 to r-1,

both results returned with probability dependent on N, $|p - 1/2|$ and probability of approximation of F function.

Let us now consider the influence of bias $\varepsilon_{ij} = p_{ij} - ½$ resulting from the use of probabilistic counting method on the success rate of the attack. A basic construction element of linear cryptanalysis with probabilistic counting method is a probabilistic approximation of non-linear operations (in DES: S-boxes). We introduce the probabilistic approximation and we will define probability with which the probabilistic approximation holds. Let $\alpha \circ \Gamma\alpha$ represent a numerical value of the vector obtained through the selection of bits from vector α chosen by non-zero positions of masking vector $\Gamma\alpha$. Example: α = [1011], $\Gamma\alpha$ = [1001], then $\alpha \circ \Gamma\alpha$ denotes numerical value which represents vector [11]: 3.

Definition 1

Let $\alpha \circ \Gamma\alpha$ denote the value of visible input bits to S-box S_i, $\alpha \circ \overline{\Gamma\alpha}$ denote the value of invisible bits. Let $\beta \bullet \Gamma\beta$ denote the value of modulo 2 sum of chosen output bits of S_i. We define the probabilistic approximation of S-box S_i ($1 \le i \le 8$), as a dependence between visible bits on the input to S_i $\alpha \circ \Gamma\alpha$ and a value of the modulo 2 sum of chosen output bits from S_i, which holds with probability p. We denote a probabilistic approximation of S_i as: $\Psi S_i(\Gamma\alpha, \Gamma\beta)$. ▫

Definition 2

For given S-box S_i (i = 1,2, .., 8), and non-zero vectors $\Gamma\alpha$ and $\Gamma\beta$ ($1 \le \Gamma\alpha \le 63$, $1 \le \Gamma\beta \le 15$), with constant $\alpha \circ \Gamma\alpha$, we define probability of probabilistic approximation as a proportion of number of events s.t. a value of modulo 2 sum of output bits from S_i chosen by $\Gamma\beta$ assumes value zero, under condition that the visible input bits to S-box S_i indicated by $\Gamma\alpha$ assume value $\alpha \circ \Gamma\alpha$, averaged over values of all invisible input bits:

$$\Pr_{0|\alpha\circ\Gamma\alpha}(\Gamma\alpha, \Gamma\beta) = \frac{1}{2^{W_H(\overline{\Gamma\alpha})}} \#\{\beta \bullet \Gamma\beta = 0 \mid \beta = S_i(\alpha)\} ▫ \tag{12}$$

Example: Let's consider S-box S_5. Let's assume that there are 4 visible input bits to S_5: $\Gamma\alpha$ = [110011], α assumes following values [000000], [000100], [001000], [001100], and $\Gamma\beta$ = [1111] then computation process of the value $\Pr_{0|\alpha\circ\Gamma\alpha}$ is illustrated by table 5. A value $\alpha\circ\Gamma\alpha$ is computed as a scalar product of vectors α and $\Gamma\alpha$, and similarly with $\beta \bullet \Gamma\beta$. Then we obtain:

$$\Pr_{0|\alpha\circ\Gamma\alpha=0}([110011], [1111]) = 0 \tag{13}$$

as a proportion of number of columns in which $\beta \bullet \Gamma\beta = 0$, to the number of all columns.

Table 5. Computing the value of $\Pr_{0|\alpha\circ\Gamma\alpha}$([110011], [1111])

α	000000	000100	001000	001100
$\Gamma\alpha$	110011	1110011	110011	110011
$\alpha \circ \Gamma\alpha$	0	0	0	0
$\alpha \circ \overline{\Gamma\alpha}$	0	1	2	3
$\beta = S_5(\alpha)$	0010	0100	0111	1101
$\Gamma\beta$	1111	1111	1111	1111
$\beta \bullet \Gamma\beta$	1	1	1	1
$\Pr_{0\|\alpha\circ\Gamma\alpha=0}$([110011], [1111])	0 / 4 = 0			

We define an average probability and an average bias of probability of probabilistic approximation $\Pr_{0|\alpha\circ\Gamma\alpha}$ ($\Gamma\alpha$, $\Gamma\beta$) computed over all possible values of visible effective input bits to the S-box:

Definition 3

For given S-box S_i (i = 1,2, .., 8) and non-zero vectors $\Gamma\alpha$ and $\Gamma\beta$ ($1 \le \Gamma\alpha \le 63$, $1 \le \Gamma\beta \le 15$) we define average probability $\tilde{p}_{\Psi S_i}(\Gamma\alpha, \Gamma\beta)$ as a proportion of sum of absolute values of biases of conditional probabilities from ½: $\Pr_{0|\alpha\circ\Gamma\alpha}(\Gamma\alpha, \Gamma\beta)$, where the sum is taken over all possible values of visible input bits ($\alpha\circ\Gamma\alpha$), to the number of all possible values of visible input bits:

$$\tilde{p}_{\Psi S_i}(\Gamma\alpha, \Gamma\beta) =$$

$$\frac{1}{2}+\frac{1}{2^{W_H(\Gamma\alpha)}}\sum_{\alpha\circ\Gamma\alpha=0}^{2^{W(\Gamma\alpha)}-1}\left|\frac{1}{2}-\Pr_{0|\alpha\circ\Gamma\alpha}(\Gamma\alpha,\Gamma\beta)\right|\square. \qquad (14)$$

Definition 4

For given S-box S_i (i = 1,2, .., 8) and non-zero vectors $\Gamma\alpha$ and $\Gamma\beta$ ($1 \le \Gamma\alpha \le 63$, $1 \le \Gamma\beta \le 15$) we define average bias of probability $\tilde{p}_{\Psi S_i}(\Gamma\alpha, \Gamma\beta)$ from ½ as:

$$\tilde{\varepsilon}_{\Psi S_i}(\Gamma\alpha, \Gamma\beta) = \tilde{p}_{\Psi S_i}(\Gamma\alpha,\Gamma\beta) - \frac{1}{2} =$$

$$= \frac{1}{2^{W_H(\Gamma\alpha)}}\sum_{\alpha\circ\Gamma\alpha=0}^{2^{W(\Gamma\alpha)}-1}\left|\frac{1}{2}-\Pr_{0|\alpha\circ\Gamma\alpha}(\Gamma\alpha,\Gamma\beta)\right|\square. \qquad (15)$$

Example

Let's consider S-box S_5. Assume that there are 4 visible bits on the input to S_5: $\Gamma\alpha$ = [110011], probabilities $\Pr_{0|\alpha\circ\Gamma\alpha}(\Gamma\alpha,\Gamma\beta)$ for all values of $\alpha\circ\Gamma\alpha$ are given in following table:

Table 6. Probability distribution $\Pr_{0|\alpha\circ\Gamma\alpha}(\Gamma\alpha,\Gamma\beta)$ as a function of values of visible input bits to S-box ($\alpha\circ\Gamma\alpha$)

$\alpha\circ\Gamma\alpha$	0	1	2	3	4	5	6	7
$\Pr_{0\mid\alpha\circ\Gamma\alpha}(\Gamma\alpha,\Gamma\beta)$	0	0	¾	¼	¼	¾	1	1
$\alpha\circ\Gamma\alpha$	8	9	10	11	12	13	14	15
$\Pr_{0\mid\alpha\circ\Gamma\alpha}(\Gamma\alpha,\Gamma\beta)$	¼	¼	0	0	1	1	¾	¾

then:

$$\tilde{\varepsilon}_{\Psi S_i}(\Gamma\alpha, \Gamma\beta) =$$

$$\frac{1}{2^4}\sum_{\alpha\circ\Gamma\alpha=0}^{2^4-1}\left|\frac{1}{2}-\Pr_{0|\alpha\circ\Gamma\alpha}([110011],[1111])\right| = 0{,}375. \qquad (16)$$

In DES maximum biases can be observed in following cases:

Table 7. Maximum values of average bias of probabilistic approximations as a function of visible input bits to a S-box in DES

number of visible bits	approximated S-box	input mask $\Gamma\alpha$	output mask $\Gamma\beta$	$\tilde{\varepsilon}_{\Psi S_i}(\Gamma\alpha, \Gamma\beta)$
1	S_5	0x10	0x0f	0,3125
2	S_5	0x11	0x0f	0,3125
2	S_5	0x12	0x0f	0,3125
2	S_5	0x14	0x0f	0,3125
2	S_5	0x18	0x0f	0,3125
2	S_5	0x30	0x0f	0,3125
3	S_0	0x16	0x0f	0,375
3	S_7	0x16	0x0f	0,375
4	S_7	0x1e	0x0f	0,4375
5	S_3	0x3e	0x0f	0,5

As we can see maximum average bias of probabilistic approximation with 1 or 2 visible bits is equal to the bias of best linear approximation.

With knowledge of probabilistic approximation we can start to construct a probabilistic round approximation in similar way as in deterministic approach and then we can construct the attack on DES. But before we do this, we sketch the method of estimation of number of texts needed in analysis. As we mentioned above the number of analysed texts depends on the probability of linear expression and on the probability of probabilistic approximation of F function. This probabilistic approximation can be treated as a random variable, which is equal to zero with probability $\tilde{p}_{\Psi S_i}$ and to 1 with probability 1-$\tilde{p}_{\Psi S_i}$. With the assumption about independence of subkeys we can use Piling-Up lemma to calculate a bias of a new linear expression as:

$$\varepsilon_r = 2 * |p - ½| * |\tilde{p}_{\Psi S_i} - ½|, \qquad (17)$$

so the number of pairs of texts needed in attack is equal to:

$$N = c * (2 * |p - ½| * |\tilde{\varepsilon}_{\Psi S_i}|)^{-2}. \qquad (18)$$

Now we can mount an attack on DES reduced to 3 rounds. Best linear expression [8] is following:

$$P^L[15] \oplus P^H[7,18,24,29] \oplus C^L[15] \oplus C^H[7,18,24,29] = K_1[22] \oplus K_3[22]. \qquad (19)$$

To implement an attack with probabilistic counting method on 3-round DES we use two round linear expression and instead of last round approximation we use a probabilistic approximation with 4 visible bits on the S-box input. Average bias of probabilistic approximation for S_5 with $W_H(\Gamma\alpha)$ = 4 and $\Gamma\beta$ = [1111] is equal to $\tilde{\varepsilon}_{\Psi S_i}$ = 0,375. So the number of texts needed in attack is equal to:

$$N = c * (2 * 20/64 * 24/64)^{-2} = c * 18{,}2. \qquad (20)$$

For comparison the number of texts needed in basic attack is equal to:

$$N = c * (2^2 * 20/64 * 1/2 * 20/64)^{-2} = c * 26{,}2. \qquad (21)$$

Moreover in a first case we determine 4 subkey bits and $K_1[22]$, and in the second case only $K_1[22] \oplus K_3[22]$.

In a basic form linear cryptanalysis with probabilistic counting method is not so effective as other extensions to linear cryptanalysis. E.g. Shimoyama's attack on 3-round DES needed only:

$$N = c * (2^2 * 20/64 * 1/2 * 1/2) = c * 10{,}2 \qquad (22)$$

pairs of texts. So probabilistic counting can be treated only as a component e.g. a component of attack with additional round reduction.

We sketch a linear cryptanalysis with reduction of two consecutive rounds of 4-round DES. We use 2-round linear expression of the following form:

$$P^H[7,18,24,29] \oplus P^L[15] \oplus C^H_2[7,18,24,29] = \Sigma_z K_z \bullet \Gamma K_z, \qquad (23)$$

and a probabilistic approximation in r-1 round of the following form: ΨS_5 (α, [110011],[1111]), which holds with average bias 0,375. Taking into account an inversion of F function in round r, we obtain:

$$P^H[7,18,24,29] \oplus P^L[15] \oplus C^H_4[7,18,24,29] \oplus \sim F_3(C^L_3, K^v_3)[7,18,24,29] \oplus F_4(C^L_4, K_4)[0,1,2,3,4,15,16,17,18,19,20,21,22,23,24,27,28,29,30,31] = \Sigma_z K_z \bullet \Gamma K_z. \qquad (24)$$

Experimentally obtained success rate is following:

Table 8. Success rate of linear cryptanalysis with reduction of two consecutive rounds on 4 round DES.

N SR	$2*(2*\|p-\frac{1}{2}\|\tilde{\varepsilon}_{\Psi S_i})^{-2}$	$4*(2*\|p-\frac{1}{2}\|\tilde{\varepsilon}_{\Psi S_i})^{-2}$	$8*(2*\|p-\frac{1}{2}\|\tilde{\varepsilon}_{\Psi S_i})^{-2}$	$16*(2*\|p-\frac{1}{2}\|\tilde{\varepsilon}_{\Psi S_i})^{-2}$
experimental	30%	60%	80%	90%
theoretical	48,6%	78,5%	96,7%	99,9%

6. CONCLUSIONS

We have implemented a linear cryptanalysis with probabilistic counting method on DES. We proposed linear cryptanalysis with two consecutive round reduction using the probabilistic counting method, which form the basis for a construction of 3R attack.

The major limitation in use of linear cryptanalysis with reduction of additional rounds is the memory complexity of an attack. But it is possible to mount 3R attack in which there are 2 outer rounds reduced and the second or one before last round. The 3R attack can be more effective that 2R attack for DES, if in second or in penultimate round the probabilistic approximation will be used which holds with average bias $\tilde{\varepsilon}_{\Psi S_i}$ ($\Gamma\alpha$, $\Gamma\beta$) bigger than 20/64 (the value of best deterministic approximation). It can happen e.g. in following cases the probabilistic approximation has W($\Gamma\alpha$) = 3 and approximated s-box is S_1 or S_8, or W($\Gamma\alpha$) = 4 and approximated s-box is S_5. In the first case minimum memory requirement is equal to $2^{3*6+6+3}$ = 128 MB, and in the second case $2^{4*6+4+6}$=16 GB. So in the first case it is possible to attack DES on the PC, while in the second one a specialised device will be needed. We can theoretically estimate that in the second case, the number of texts will be reduced to 69% of texts used in 2R attack. We can achieve further improvement by combining the proposed attack with other extensions.

7. FURTHER RESEARCH

Our further research will concentrate on combining extensions of linear cryptanalysis with proposed 3R attack. Also our attention will be concentrated on explaining the dependence of success rate of attack on the distribution of probabilities in the inverted S-box.

REFERENCES

1 Biham, E., A. Shamir, 1993. "Differential Cryptanalysis of Data Encryption Standard", Springer Verlag, 1993.

2 U. Blöcher, M. Dichtl, „Problems with the Linear Cryptanalysis of DES Using more than one Active S-Box per Round", Fast Software Encryption, Springer Verlag 1994, ISBN 3-540-60590-8.

3 Data Encryption Standard. Federal Information Processing Standard (FIPS), Publication 46, National Bureau of Standards, U.S., Department of Commerce, Washington D.C., January 1977.

4 C. Harpes, G. G. Kramer, J. L. Massey, 1995. "A Generalisation of Linear Cryptanalysis and Applicability of Matsui's Piling-Up Lemma", Advances in Cryptology Eurocrypt'95.

5 B. S. Kaliski Jr., M. J. B. Robshaw, „Linear Cryptanalysis Using Multiple Approximations", Advances in Cryptology Crypto'94, Springer Verlag 1994, ISBN 3-540-58333-5.

6 L. R. Knudsen, M. J. B. Robshaw, „Non-Linear Approximations in Linear Cryptanalysis, Advances in Cryptology Eurocrypt'96, Springer Verlag 1996, ISBN 3-540-61186-X.

7 S. Langford, M. E. Hellman, „Differential-linear Cryptanalysis", Advances in Cryptology Crypto'94, Springer Verlag 1994, ISBN 3-540-58333-5.

8 M. Matsui, „Linear Cryptanalysis Method for DES Cipher", Advances in Cryptology Eurocrypt'93.

9 M. Matsui, „On Correlation Between the Order of S-boxes and the Strength of DES", Advances in Cryptology Eurocrypt'94, Springer Verlag 1994, ISBN 3-540-60176-7.

10 M. Matsui, „The First Experimental cryptanalysis of Data Encryption Standard", Advances in Cryptology Crypto'94, Springer Verlag 1994, ISBN 3-540-58333-5.

11 K. Ohta, S. Morai, K. Aoki, „Improving the Search Algorithm for Best Linear Expression", Advances in Cryptology Crypto'95, Springer Verlag 1995, ISBN 3-540-60221-6.

12 K. Sakurai, S. Furuya, "Improving linear cryptanalysis of LOKI91 by probabilistic counting method", Fast Software Encryption Workshop (FSE4), Haifa, Israel, 1997.

13 T. Shimoyama, T. Kaneko, "Quadratic Relation of S-Box and Its Application to the Linear Attack of Full Round DES", Advances in Cryptology, Crypto'98. ISBN 3-540-64892-5.

14 A. Zugaj, „Algorithms for finding linear expressions", (in Polish), IV Polish National Conference on the Applications of Cryptography ENIGMA'2000, Mai 2000.

15 A. Zugaj, K. Górski, Z. Kotulski, A. Paszkiewicz, J. Szczepański, S. Trznadel, "Linear cryptanalysis of DES algorithm", (in Polish), seminar notes Institute of Telecommunications, Warsaw University of Technology, April 1998.

16 A. Zugaj, K. Górski, Z. Kotulski, A. Paszkiewicz, J. Szczepański, S. Trznadel, „Linear cryptanalysis", (in Polish) PWT, December 1998.

17 A. Zugaj, K. Górski, Z. Kotulski, J. Szczepański, A. Paszkiewicz, "Extending linear cryptanalysis – theory and experiments", Regional Conference on Military Communication and Information Systems, RCMCIS'99, October 6-8, 1999.