Detection of Stegosystems Using Block Ciphers for Encryption of the Embedded Messages

Valery Korzhik, Ivan Fedyanin, Nguyen Duy Cuong
The Bonch-Bruevich Saint-Petersburg State University of Telecommunications
Saint-Petersburg, Russia
val-korzhik@yandex.ru,{ivan.a.fedyanin, cuong0111}@gmail.com

***Abstract*—We consider firstly stegosystems in which the embedded messages are encrypted preliminary by any block cipher in a codebook mode. Detection of stegosystems presence is performed if the number of the repeated extracted blocks exceeds of some given threshold. Experiments demonstrate that if the embedding data are meaningful text and extraction algorithm is known for the attacker, then a distinguishing between stego and cover objects occur very reliable even for matrix embedding with very small rate . If the embedding data are known for attacker then it is used attack based on a calculation of mutual information between message and the encrypted data with application of k-nearest neighbor distance. Experiments show that for not very strong ciphers with block length at most 32 bits this attack is successful.**

I. Introduction

Steganalysis is a complementary task of steganography. It is well known [1] that the main goal of steganalysis (SGA) is to distinguish between cover objects (CO) and stego objects (SG) with probability better than random guessing. It is common to consider steganography in digital media where CO are both digital motionless and video images and signals like speech and music. But our experiments will be restricted for simplicity reason by motionless grey scale images only. (In the future our proposals can be extended to other types of CO without significant difficulties.)

Steganalysis is very important on two reasons. Firstly it is used as a notion that should be taken into account during design of any steganographic algorithm because such algorithm is useless if it can be easily detected by some known steganalytic method. Secondly, SGA has its own rights. In fact, it is very important to prevent a leakage of sensitive information outside of some areas because this can be arranged by steganographic methods. (It is well known system "Digital Leakage Prevention" (*DLP*) that has to provide impossibility to transmit sensitive information outside of some company area. But without steganalysis it works unwell.)

All stegosystems known before were subjected preliminary by methods of SGA. One of the first papers devoted to *SGA* was [2] but in a more complete form *SGA* has been presented in monography J. Fridrich [1]. Following to the last book one can divide methods of SGA in two main parts: targeted SGA and blind SGA. For the first part the features in SGA are constructed to a specific embedding method. The goal of blind staganalysis is to detect any steganographic method irrespectively to its embedding mechanism. It seems today that the best method of blind SGA is to use Support Vector Machines (SVM) which is realized in two stages. The first is training one on both CO and SG databases. (It is worth to note that although embedding mechanism can be unknown for steganalytic but it is possible to test many SG using embedding algorithms like "black boxes".) During the first stage some features have to be extracted both from CO and SG. At the second stage these features are used for a recognition of some new test object belonging to one of two classes: CO or SG. Algorithm of such classification is detailed in [1].

Kerckhoff assumption known in cryptography [3] can be extended also to steganography. This means that "attacker" (say steganalytic) may know all about embedding and extraction algorithm except of crypto and stego keys. The stego key usually determines a pseudo-random path through the CO where the message bits are embedded. A weak stego key creates an undetectability weakness that can be used by an attacker to extract the embedded "message" and take a decision about a presence of SG if the extracted "message" is meaningful (See Algorithm 10.2 in [1]). Moreover it is a great risk to hide the embedded message content only with the use of stego key [4]. Therefore it is required as a rule to use also very strong ciphers for message encryption.

In the book[1] (See Section 10.7) it was written that if the message was encrypted prior to embedding, then attacker cannot reliably distinguish between a random bit stream and an encrypted message.

We disagree with such conclusion and our contribution consists in a demonstration that under some conditions stegosystems can be reliably detected against covers.

In Section II we consider a scenario where it is used any strong cipher in codebook mode. In Section III it is presented a scenario where an attacker knows the exact message but cipher is not very strong. This case is application to steganography an attack known in cryptography as chosen plaintext attack (CPA), breaking of cipher semantic security, in other words. Section IV concludes the paper.

II. Steganalysis based on the use of any block cipher for message encryption in codebook mode

Let us consider any stegosystem (SG) where block cipher but in codebook mode was used for encryption of the embedded messages. We assume that in line with Kerkhoffs principle extraction algorithm is known for attacker. Even so

some stego key was used for a determination of a pseudo-random walk through the CO where the message bits are embedded, this key can be somehow found. (See [1], [4] for detail). Thus an attacker can see the ciphertext (in the case of SG presence) or some bits of cover(in the case of SG absent).

In order to distinguish these cases we execute the following property of the codebook cipher mode: *a repetition of plaintext blocks results in a repetition of corresponding ciphertext blocks.*

Lets us consider for definiteness sake SG with matrix embedding based on Hamming codes[1], although such approach can be applied to any algorithm with known extraction algorithm.

The binary Hamming codes can be determined uniquely by their $p \times 2^p - 1$ check matrix, that consists from all possible nonzero binary columns of the length p [5]. In order to embed some given message into the grey scale digital image it is necessary to extract from this image all LSBs, divide the obtained binary string on blocks of the length $2^p - 1$ and "embed" p bits of the message into each block changing only one symbol on the opposite one performing the following steps[1]:

1) Compute vector $z = xH^T \oplus m$, where x – LSB vector of the length $2^p - 1$, m –part of the message vector of the length p, T – symbol of matrix transposition
2) Find the number i of matrix H column that equals to vector z.
3) Invert the i-th symbol of x to opposite one that results in block y with embedding of the first p message symbols.

In a similar manner should be embedded next p bits of message into the next block of the length $2^p - 1$ and so on up to the end of the full message sequence. In order to extract message bits m from each block y it is necessary to do the following:

$$\mathrm{m} = \mathrm{Hy} \qquad (1)$$

Detection algorithm extracts blocks of bits by (1) and compare the number of the repeating blocks with some thresholds. If this value exceeds chosen threshold then is taken a decision that SG is found, otherwise CO is detected.

In Table I are presented parameters of matrix embedding with Hamming codes depending on the main Hamming code parameter p and for motionless grey scale images of size 512x512 pixels.

In Table II are presented the results of block repetition distribution for 240 different English texts encrypted by DES cipher and embedded into image of size 512x512 with Hamming code having p = 3.

We can see from Table II that if English texts be chosen uniformly and threshold selected as 2, then the probability to take a correct decision be about (240-1)/240 ≈ 0.995. (Of course, we have got much more experiments for different texts and parameters p but a presentation only one Table is owing paper size limitation). It is worth to note that results of experiments do not depend from images because we assume that encrypted messages are extracted correctly for any image. On the other hand the probability of false alarm (when the threshold is exceeded for covers) has to depend on images. The results of block repetition testing for 15 covers obtained by formula (1) with parameter p = 3 are presented in Table III.

TABLE I. PARAMETERS OF MATRIX EMBEDDING BY HAMMING CODES IN MOTIONLESS IMAGES OF SIZES 512X512 PIXELS DEPENDING ON THE MAIN CODE PARAMETER P

p	1	2	3	4	5	6	7	8	9	10	11	12	13	14
The length of code blocks 2^p - 1	1	3	7	15	31	63	127	255	511	1023	2047	4095	8191	16383
The length of the embedded bits	262144	174762	112347	69904	42280	24966	14448	8224	4617	2560	1408	768	416	224

TABLE II. DISTRIBUTION OF BLOCK REPETITION FOR DIFFERENT ENGLISH TEXTS ENCRYPTED BY DES CIPHER AND THE USE OF HAMMING CODES WITH PARAMETER P = 3

Number of English text	1	2	3	4	5	…	239	240
Number of block repetition	0	1	2	3	3	…	213	273

TABLE III. THE NUMBER OF 64-LENGTH BLOCK REPETITION FOR DIFFERENT IMAGES AND PARAMETER P = 3

Images	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Total number of 64-length block	1755	1755	1755	1755	1755	1755	1755	1755	1755	1755	1755	1755	1755	1755	1755
Number of repetition	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0

We can see from this table that for all images except for image 7 we get nothing repetitions. As far as image №7 it was the image with close to uniform histogram. But of course such type of images cannot be taken as a cover for steganographic embedding because its detection occurs obviously.

If we model the sequence of bit obtained by (1), as i.i.d with equal probabilities zeros and ones then we can use asymptotic Feller's formula proved for birthday paradox [6]. In fact, if we let that the number of balls in urn is 2^n where n – is the block length and the number of ball extraction is N, then in line with Feller formula we get the probability at least of two extractions (with replacing) of balls with equal number is

$$p(N,n) = 1 - \exp(-\frac{N^2}{2^{n+1}}) \qquad (2)$$

In our case n = 64 and N = 1755 (in Table III), then the probability of at least of two block repetitions be very small. Thus, it is no wonder that we get in Table III nothing repetitions for all images except of image 7th. The last case is a consequence of incorrect model taken before.

It is worth to note that steganalytic method presented above does not work for such cipher modes as the cipher-block chaining mode (CBC) and the cipher feedback mode (CFB) because as it well known [3] repetition of plaintext blocks does not result always in a repetition of ciphertext blocks. But we get important for a steganography conclusion: *encryption of the embedding messages must be provided either by CBC or CFB cipher modes but never by codebook mode.*

III. Steganalysis for the case of known plaintext embedding into covers

This scenario is, of course, comparatively uncommon but information security is very important area to be avoidable even rare situations. The more, in cryptography it is commonly to consider chosen-plaintext attack that tries to break so called *semantic security* [7]. In steganography this scenario assumes that the embedded message is encrypted by some block cipher and this message is known completely but it is open problem if this message is embedded or not in a given cover object?

In Fig 1 is presented a scheme of such cipher and in Table IV and Table V are presented S-box transforms and permutation mapping. Although this cipher has $2^{80} \approx 1.2 \cdot 10^{24}$ secret keys and hence a brute force attack by key exhaustion is untractable, this cipher can be easily broken, for the thing, by linear or differential cryptanalysis.

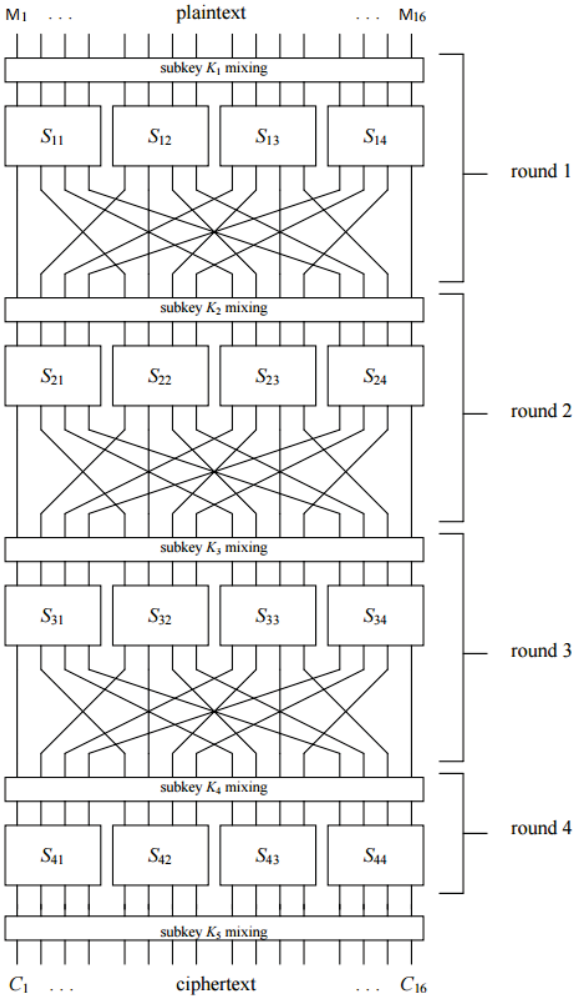

Fig. 1. Substitution-permutation block cipher with block length 16 and four rounds.

However, sometimes in stegosystems can be applied sufficiently simple encryption algorithms. Moreover we present also some extension of SPC with block length 32.

For the case of substitution-permutation cipher with block length 32 bits we extend the 16 bit cipher with addition of the second halve of scheme to the first one keeping previous transforms in S-boxes and changing Table V for permutation mapping to Table VI showed below.

TABLE IV. S-box transforms for all S-boxes and they are presented in hexadecimal system.

Input	0	1	2	3	4	5	6	7	8	9	10(A)	11(B)	12(C)	13(D)	14(E)	15(F)
Output	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

TABLE V. Permutation mappings for all cipher rounds

Input	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Output	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

TABLE VI. PERMUTATION MAPPINGS FOR 32-BIT BLOCK LENGTH CIPHER.

Input	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Output	1	5	9	13	17	21	25	29	2	6	10	14	18	22	26	30

Input	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Output	3	7	11	15	19	23	27	31	4	8	12	16	20	24	28	32

It is well known inequality for mutual information between plaintext and ciphertext that should be valid for any cryptosystem[8],[9]:

$$I(M^N, C^N) \geq H(M^N) - H(K^L) \quad (3)$$

where M^N is a sequence of message symbols of the length N, C^N is a sequence of ciphertext symbols of the length N (without of the generality lost we believe that these lengths are equal one to another), K^L is the binary key string of the length L.

We can transform inequality (1) by dividing both its sides on N:

$$I'(M^N, C^N) \geq H'(M^N) - H'(K^L) \quad (4)$$

where symbol " ′ " means that we consider a normilized to N corresponding values.

Since for computationally secure contemporary block ciphers the length of the key L is much less than the length of the message, we get asymptotically (as $N \to \infty$):

$$I'(M^N, C^N) \sim H'(M^N) > 0 \quad (5)$$

It follows from inequality (5) that if some message M^N has been in fact encrypted into ciphertext C^N with any unknown key K^L of the limited length L then for very large message length N we get nonzero mutual information $I'(M^N, C^N)$ but this value approaches to zero if M^N is not encrypted as C^N with some key. Hence we can take a decision about a choice of message that is encrypted into given ciphertext comparing the value $I'(M^N, C^N)$ with some threshold.

But the following problem appears – how it is possible to calculate mutual information $I'(M^N, C^N)$? Solution to this problem based on "binning" [10] was very hard generally but relatively recent has been published the paper [11] where it was used a method based on the notion of *k-nearest neighbour distance*. This approach can be termed as *fast mutual information calculation(FMIC)* between two N-dimension random vectors *X* and *Y*. It has been proved in [12] that FMIC can be performed by the following algorithm:

$$I(X,Y) = \Psi(1) - \left\langle \Psi(n_x + 1) + \Psi(n_y + 1) \right\rangle + \Psi(N) \quad (6)$$

where $X = \{x_1, x_2, ..., x_N\}, Y = \{y_1, y_2, ..., y_N\}$ vectors corresponding to M^N and C^N, $\Psi(x)$ is digamma function, $\Psi(x) = \Gamma(x)^{-1} d\Gamma(x)dx$ that satisfies the recursion $\Psi(x+1) = \Psi(x) + 1/x$ and $\Psi(1) = C$, where $C = 0.5772156...$ is the Euler-Mascheroni constant. For large x, $\Psi(x) \approx \log x - 1/2x$. $n_x(i)$ is the number of points x_j whose distance from x_i is strictly less than $\varepsilon(i)/2$ and similarly for y instead of x. Here $\varepsilon(i)/2$ is the distance from $z_i = (x_i, y_i)$ to its neighbour and $\varepsilon_x(i)/2$ and $\varepsilon_y(i)/2$ are distances between the same points projected into the X and Y subspaces. Obviously, $\varepsilon(i) = \max(\varepsilon_x(i), \varepsilon_y(i))$. $\langle ... \rangle$ is symbol that denotes an averaging both over all $i \in [1,...,N]$ and over all realizations of random samples. But in our case we average only on all samples $i \in [1,...,N]$ that is $\langle ... \rangle = \frac{1}{N} \sum_{k=1}^{N} (...)$.

In order to implement relation (6) for estimation of left side inequality (5) we map each of plaintext blocks $M_i = (m_{i1}, m_{i2}, ..., m_{in})$ into one integer X_i and each of the ciphertext blocks $C_i = (c_{i1}, c_{i2}, ..., c_{in})$ into one integer Y_i following trivial relations, respectively:

$$X_i = \sum_{j=0}^{n-1} x_{ij} 2^j, Y_i = \sum_{j=0}^{n-1} y_{ij} 2^j, i = 1,2,...,N \quad (7)$$

where n is the block cipher length, x_{ij}, y_{ij} binary symbols of plaintext M^N and ciphertext C^N, respectively. (We assume of course that block cipher is binary and has the same length n of input and output blocks).

Experimental investigation of the technique described above is presented as follows.

We generate pseudo randomly two binary sequences M_{I} and M_{II} both of the length $n \cdot N$, where $n = 16$ is the cipher block length and N is the number of tested blocks. One of these sequences, say M_{I} is encrypted by Heye's block cipher that gives $n \cdot N$ ciphertext bits. (It is worth to noting that in the case of meaningful plaintext the entropy $H'(M^N)$ in (4) be lesser than for truly random binary sequence but it be still nonzero. Hence the proposed method works but we should select plaintext as close to truly random one only for simplicity reasons). Next we calculate mutual information

$I(M_{\mathrm{I}},C)$ by (6) and (7), where X_i are integers corresponding to M_I and Y_i are integers corresponding to $C = f(M_{\mathrm{I}},K)$, where $f(.)$ is the encryption function for Heye's 16-bit block cipher with 80-bit key chosen pseudo randomly. After that it is calculated also by (6) and (7) mutual information between ciphertext C obtained after encryption of plaintext M_{I} and independent on it another plaintext M_{II}. The results of such calculations against the number of message bits N are presented in Table VII.

TABLE VII. MUTUAL INFORMATION BETWEEN CIPHERTEXT AND PLAINTEXT CORRESPONDING AND NO CORRESPONDING TO GIVEN CIPHERTEXT AGAINST THE PLAINTEXT BIT LENGTH N.

N	10^2	10^3	10^4	2×10^4	4×10^4	8×10^4	3×10^5	10^6
$I(M_{\mathrm{I}},C)$	0.3	1.2	5.52	7.057	8.77	10.3	12.65	14.24
$I(M_{\mathrm{II}},C)$	-0.09	0.053	0.03	0.04	0.08	0.13	0.373	0.89

We can see from this Table VII that in fact mutual information $I(M_{\mathrm{I}},C)$ for valid plaintext M_I encrypted into C increases with increasing of N and approaches to normalized entropy of truly random binary string of the length 16. Mutual information $I(M_{\mathrm{II}},C)$ between ciphertext (obtained for plaintext M_{I}) and plaintext M_{II} is close to 0. It is sufficiently to select some threshold in order to distinguish between valid and invalid plaintexts for given ciphertext already for $N \geq 10^3$.

In Table VIII are presented results of calculation for cross correlation $R(C,M)$ between sequence C and sequences M_{I} and M_{II} which show that such criteria cannot be used for a breaking of block cipher semantic security. (This is a consequence of course, a presence of nonlinear transforms in algorithm of Heye's block cipher containing into its S-boxes.)

TABLE VIII. CROSS CORRELATION BETWEEN CIPHERTEXT C AND PLAINTEXTS M_I, M_{II} AGAINST THE PLAINTEXT BIT LENGTH N..

N	4×10^4	8×10^4	3×10^5	10^6
$R(M_{\mathrm{I}},C)$	-0.000680	-0.038	0.011	-0.000977
$R(M_{\mathrm{II}},C)$	-0.0019	-0.000556	0.003	-0.00016

We consider next block cipher with the same structure as Heye's cipher but with block length 32 and with round keys consisting from 32 bit each. S-box transforms are shown in Table IV and permutation mapping is shown in Table VI. Experiment with such "extended cipher" was arranged similarly as for ordinary cipher described before with only differences that two plaintexts M_{I} and M_{II} have the length 32 bits and the same length has ciphertext C. The results of simulations are presented in Table IX.

TABLE IX. MUTUAL INFORMATION BETWEEN CIPHER TEXT AND PLAINTEXT CORRESPONDING AND NOT GIVEN CIPHER TEXT AGAINST THE PLAINTEXT BIT LENGTH N.

N	10^3	10^4	2×10^4	4×10^4	8×10^4	3×10^5	10^6
$I(M_{\mathrm{I}},C)$	-0.065	0.025	0.038	0.078	0.083	0.3626	0.976
$I(M_{\mathrm{II}},C)$	-0.03	-0.007	0.0025	-0.012	0.0055	0.0014	0.0017

We can see from Table IX, that despite of the fact that mutual information $I(M_{\mathrm{I}},C)$ grows much slower with increasing of N than similar value for 16-bit block cipher (see Table VII) it is still exceeds the value $I(M_{\mathrm{II}},C)$ where $N \geq 10^4$. This means that after a choice of appropriate threshold it is possible to distinguish "valid" plaintext from "invalid" one for a given ciphertext. Thus the proposed approach can break semantic security of at least for block ciphers with limited block length $n \leq 32$.

Our experiments with DES block cipher having block length 64 bits showed that this problem is rather untractable at least with the use of ordinary PC.

IV. CONCLUSION

In this paper, two new steganalytic algorithms are proposed. The idea of the first one is to investigate the number of repeating blocks after application of extraction algorithm. If this number exceeds some threshold then was taken a decision about SG presence, otherwise about presence of cover object. We showed that the proposed method works well if for encryption of the embedded messages has been used any block cipher but only in codebook mode, and the extraction algorithm is known or can be found. We get also important conclusion that in order to prevent such attack it is necessary to use either CBC or CFB but never codebook mode.

The second stegoanalytic algorithm can be used only if the embedding plaintext is known in advance and it is necessary to prove that this plaintext after encryption was embedded namely into testing stegotext. This steganographic attack coincides with known in cryptography chosen-plaintext attack. But technique of this algorithm implementation is relatively new and it is based on a calculation of mutual information between plaintext and ciphertext. In order to compute this mutual information we execute fast k-nearest neighbor distance method proposed recently by A. Krasko et al.

Unfortunately, we were able to realize this algorithm for sufficiently weak block ciphers with block length of 32.

It is worth to note that in paper [12] we propose the third new stegoanalytic method based on estimation of pseudorandomness for the extracted information. This approach works well for any block cipher, any cipher modes and for many stegoalgorithms.

REFERENCES

[1] J. Fridrich “Steganalysis in Digital Media”, Cambridge University Press, 2010

[2] N.F. Johuson and S. Jajodia “Steganalysis: The investigation of hidden information. In proceedings IEEE Information Technology Conference, Syracuse, NY, 1998

[3] Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone, “Handbook of Applied Cryptography”,CRC Press, 1996

[4] J. Fridrich et al. “Forensic steganalysis: Determining the stego key in spatial domain steganography” In Proc. SPIE VII volume 5681, p.631-642, 2—5

[5] F. J. M. Williams and N. J. Sloane, The Theory of Error-correcting Codes, North-Holland, Amsterdam,1977

[6] W. Feller, “An introduction to probability theory and its applications” YWS, New-York, 3rd edition, 1988.

[7] A more powerful adversary. Security against chosen-plaintext attacks. Computer Science Department, Wellesley College (http://cs.wellesley.edu/~cs310/lectures/07_CPA_slides_handouts .pdf)

[8] C.E. Shannon “Communication theory of secrecy system”, Bell System Technical Journal 28, P. 650-715, 1949

[9] C. Henck van Tillory “Fundamentals of Cryptography”, Kluwer Acad. Publisher, 2000

[10] A. Hyvariach, J. Karpunen and E. Oja, “Independent Component Analysis”, Willey, 2001

[11] A. Kraskov, H. Stogbauer and P. Grassberger “Estimating mutual information”, Physical Review, E69, 066138, 2004.

[12] V. Korzhik, I. Fedyanin, A, Godlewski, G. Morales Luna, “Steganalysis based on statistical properties of the encrypted messages”, MMM-ACNS-2017 (Submitted)