Key Classification Attack on Block Ciphers

Maghsoud Parviz, Math. Dept., Sharif University of Technology,Tehran, IRAN
maghsoudparviz@alum.sharif.ir

Seyed Hassan Mousavi, Math. Dept., Isfahan University of Technology, Isfahan, IRAN
shnmousavi_iut@yahoo.fr

Saeed Mirahmadi, Pooyesh Educational Institute, Qom, IRAN

***Abstract*— In this paper, security analysis of block ciphers with key length greater than block length is proposed. For a well-designed block cipher with key length k and block length n s.t. k>n and for all P, C, there are 2^{k-n} keys which map P to C. For given block cipher, if there is an efficient algorithm that can classify such keys, we propose an algorithm will be able to recover the secret key with complexity $O\left(max\left\{2^{n}, 2^{k-n}\right\}\right)$. We apply this method on 2-round block cipher KASUMI.**

Keywords- Block cipher, Key classes, key length, block length, KASUMI.

I. INTRODUCTION

In design of block cipher, one step is to determine key length and block length. There are some standard block ciphers such as KASUMI , IDEA, 3DES, AES-256 which their block length is smaller than key length. Because of pseudo-randomness property of these ciphers, their key space can be portioned into equivalence classes of keys with the same number of elements. For instance in block cipher KASUMI, the key space would have 2^{64} classes that each contains 2^{64} different keys.

In our proposed attack, we assume that there is an efficient algorithm which computes the equivalence class of an arbitrary key w.r.t. a plaintext P.

We present an efficient algorithm for computing equivalence class of arbitrary key for one-round and 2-round KASUMI. Hence, in the rest, we give a short introduction to KASUMI block cipher.

KASUMI is modified version of the block cipher MISTY1 [1], which is optimized for hardware performance. In the past few years, it has received a lot of attentions from the cryptographic researchers. Kuhn introduced the impossible differential attack on 6-round KASUMI with data complexity 2^{55} and time complexity 2^{100} [2], which has been recently extended to 7-round KASUMI[3]. This attack on the last 7 rounds needs $2^{114.3}$ encryptions with $2^{52.5}$ chosen plaintexts and on the first 7 rounds, the data complexity is 2^{62} known plaintexts and the time complexity is $2^{115.8}$ encryptions.

In 2007, a higher order differential attack has been published on 5-round KASUMI with data complexity $2^{28.9}$ and time complexity $2^{31.2}$ [4].

Since the key schedule of KASUMI is linear, many related-key attacks have published. A related-key differential attack on 6-round KASUMI has presented in 2002[5]. The first related-key attack on the full 8-round KASUMI was proposed by Biham et al.with $2^{76.1}$ encryptions [6], which was improved to a practical related-key(sandwich) attack on the full KASUMI by Dunkelman et al. with data complexity 2^{26} and time complexity 2^{32} [7]. However, assumption for these kinds of attack is controlling over the differences of two or more related keys. With this assumption, resulting attacks isn't applicable in most real-world usage scenarios [8].

In this paper, we propose a new attack which isn't based on differential type attacks.

In this paper, first we introduce our new method and then apply it to reduced-round KASUMI. In section II, we introduce details of our method. In section III, the method has applied to KASUMI. In section IV, we propose an algorithm for generating equivalence classes of KASUMI keys. Finally, after experimental results, we will give some suggestions and recommendations for future works.

II. OUR PROPOSED ATTACK

Consider a block cipher $E:\{0,1\}^{n}\times\{0,1\}^{k}\rightarrow\{0,1\}^{n}$ in which n is block length and k is key length.

For a well-designed block cipher with key length k and block length n s.t. k>n and for all P_0, C_0, there are 2^{k-n} keys K which

$$E\left(P_0,K\right)=C_0$$

Let $P_0\in\{0,1\}^{n}$, we can show that the following relation is an equivalence relation.

$$K\sim K' \text{ iff } E\left(P_0,K\right)=E\left(P_0,K'\right)$$

In the rest of this paper, $[K]_{P_0}$ stands for equivalence class of key $K\in\{0,1\}^{k}$. For block cipher E, suppose that there exists an efficient algorithm Γ which computes the equivalence class of an arbitrary key w.r.t. plaintext P_0.

Now we describe our new attack. Let K_0 is the secret key of the system and we have $r := \left\lceil \frac{k}{n} \right\rceil$ pairs $(P_i, C_i), 0 \le i \le r-1$, s.t. $E(P_i, K_0) = C_i$.

Algorithm 1:

```
1. Select the key K such that $E(P_0, K) = C_0$.
2. Find key $K' \in [K]_{P_0}$ s.t for $0 \le i \le r-1$, $E(P_i, K') = C_i$.
```

Based on pseudo-randomness property of encryption algorithm, it is expected that there is exactly one key such that $E(P_0, K) = C_0$ for every interval $\lfloor i \times 2^n, (i+1) \times 2^n \rfloor$ and $0 \le i \le 2^{k-n}$. For finding a key K such that $E(P_0, K) = C_0$, it would be sufficient to look up entire keys within $\lfloor 0, 2^n - 1 \rfloor$. It can be efficiently done using rainbow-tables and time-memory-data-trade-off methods with complexity lower than 2^n.

Also because $K_0 \in [K]_{P_0}$, we generate the class $[K]_{P_0}$ by Γ algorithm. If a key in this class can correctly decrypts all $\left\lceil \frac{k}{n} \right\rceil$ pair (P_i, C_i), select it as main unknown key.

III. KASUMI BLOCK CIPHER

In this section, we introduce briefly the general structure and properties of KASUMI as well as functions that used in KASUMI.

A. KASUMI block cipher

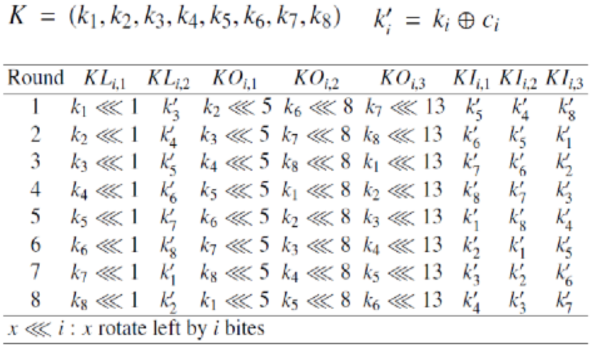
KASUMI is a block cipher used for the security of 3GPP systems such as UMTS, GSM and GPRS. Both the confidentiality (f8) and integrity function (f9) in UMTS are based on KASUMI. In GSM, KASUMI is used in the A5/3 **a**lgorithm for generating key stream and in GPRS in the GEA3 key stream generator.

KASUMI is an 8-round Feistel block cipher algorithm; it operates on 64 bit input to produce 64 bit output under a 128-bit key. In each round, there are two functions: the FO function which is a 3-round, 32-bit Feistel structure, and the FL function which receives 32-bit as input and produces 32-bit output. The order of using two mentioned functions in the cipher is affected by the round number. In the odd round, the first function is FL and in the even round, the FO function is used first.

The FO function also uses four-round Feistel FI function in a recursive structure. The FI function receives 16-bit as input and produce 16-bit as output. It uses two S-boxes S7 (7-bit to 7-bit permutation) and S9 (9-bit to 9-bit permutation).

Using a simple key schedule of figure 1, the round function uses a round key which consists of eight 16-bit sub keys derived from the original 128-bit key. The FL function uses 32-bit sub-keys $KL_{i,j}$ in round i where j=1 or 2. The FO function uses 96-bit sub-keys $KO_{i,j}$ and $KI_{i,j}$.

The 128-bit key *K* is divided into eight 16-bit sub keys K_i :



$K = (k_1, k_2, k_3, k_4, k_5, k_6, k_7, k_8) \quad k'_i = k_i \oplus c_i$

Round	$KL_{i,1}$	$KL_{i,2}$	$KO_{i,1}$	$KO_{i,2}$	$KO_{i,3}$	$KI_{i,1}$	$KI_{i,2}$	$KI_{i,3}$
1	$k_1 \lll 1$	k'_3	$k_2 \lll 5$	$k_6 \lll 8$	$k_7 \lll 13$	k'_5	k'_4	k'_8
2	$k_2 \lll 1$	k'_4	$k_3 \lll 5$	$k_7 \lll 8$	$k_8 \lll 13$	k'_6	k'_5	k'_1
3	$k_3 \lll 1$	k'_5	$k_4 \lll 5$	$k_8 \lll 8$	$k_1 \lll 13$	k'_7	k'_6	k'_2
4	$k_4 \lll 1$	k'_6	$k_5 \lll 5$	$k_1 \lll 8$	$k_2 \lll 13$	k'_8	k'_7	k'_3
5	$k_5 \lll 1$	k'_7	$k_6 \lll 5$	$k_2 \lll 8$	$k_3 \lll 13$	k'_1	k'_8	k'_4
6	$k_6 \lll 1$	k'_8	$k_7 \lll 5$	$k_3 \lll 8$	$k_4 \lll 13$	k'_2	k'_1	k'_5
7	$k_7 \lll 1$	k'_1	$k_8 \lll 5$	$k_4 \lll 8$	$k_5 \lll 13$	k'_3	k'_2	k'_6
8	$k_8 \lll 1$	k'_2	$k_1 \lll 5$	$k_5 \lll 8$	$k_6 \lll 13$	k'_4	k'_3	k'_7

$x \lll i$: x rotate left by i bites

Figure 1. KASUMI key schedule

We start with FI function which is the nonlinear part of cipher.

B. FI function properties

One of the main functions used in KASUMI block cipher is

$$FI : \{0,1\}^{16} \times \{0,1\}^{16} \to \{0,1\}^{16}$$

As figure 4 shows, there are two sboxes S7 (7-bit to 7-bit permutation) and S9 (9-bit to 9-bit permutation). In the following, we give some properties for the FI function.

1. $FI(x, KI) = FI_2(KI \oplus FI_1(x))$
   where FI_1 is above part of the FI function before the key KI affects and FI_2 is below part of the FI function after the key KI affects.

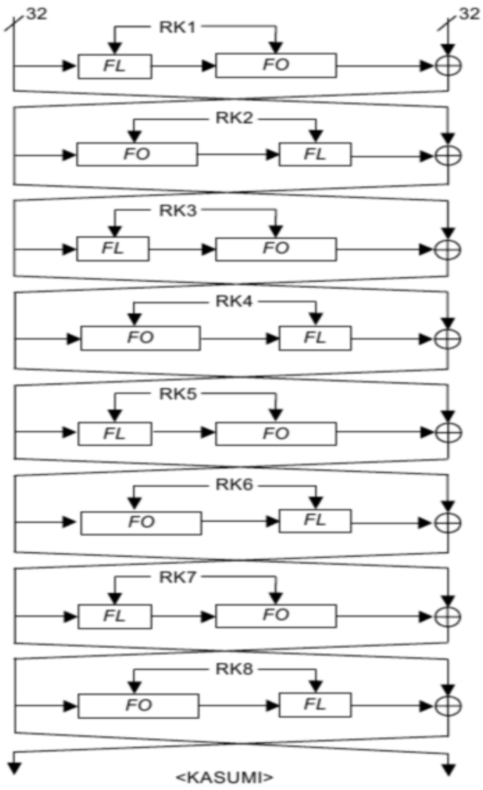


Figure 2. KASUMI block cipher

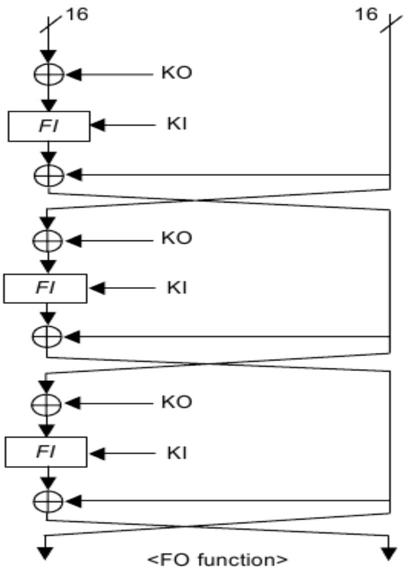


Figure 3. FO function

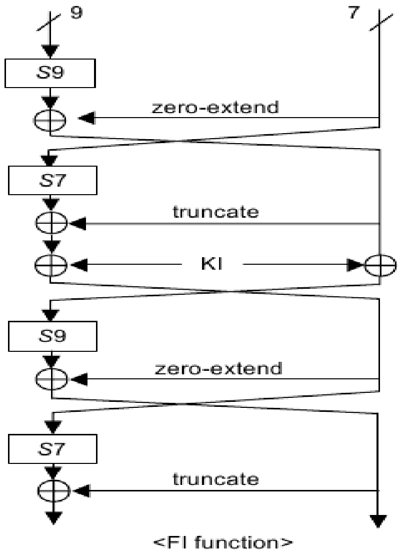


Figure 4. FI function

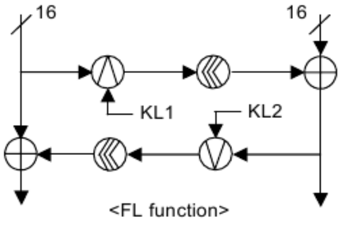


Figure 5. FL function

It is clear that FI_1 and FI_2 are invertible functions that are independent from key.

2. Suppose that

$$FI(x,KI) = y \qquad (1)$$

Where x, y and KI are input, output and key. Having two members of the set {x,y,KI}, the third one is uniquely computed efficiently. In fact, if x, KI (y, KI) are known, using encryption (decryption) procedure, y (x) is efficiently computable.

If x, y are known, then

$$KI = FI_1(x) \oplus FI_2^{-1}(y)$$

3. $\left|\left\{(x,KI); FI(x,KI) = y\right\}\right| = 2^{32}$

For all $y \in \{0,1\}^{16}$.

4. For all $x, x' \in \{0,1\}^{16}$, we can find KI and KI' s.t.

$$FI(x,KI) = FI(x',KI')$$

In fact, it is sufficient to find keys such that

$$KI \oplus KI' = FI_1(x) \oplus FI_1(x') \quad (2)$$

It is clear that the number of keys which (2) holds are 2^{16}.

5. For all $x, x', y \in \{0,1\}^{16}$, there are some keys KI, KI' s.t.

$$FI(x,KI) = FI(x',KI') = y$$

Using equation (2), these keys can be easily computed.

C. FO function properties

Function $FO: \{0,1\}^{32} \times \{0,1\}^{96} \to \{0,1\}^{32}$ is the main part of KASUMI.

Suppose that $X = (X_L, X_R)$ and $Y = (Y_L, Y_R)$ are input and output of FO function, then

$$Y_L = X_R \oplus FI(X_L \oplus KO_1, KI_1) \oplus FI(X_R \oplus KO_2, KI_2) \quad (3)$$

$$Y_R = Y_L \oplus FI(X_R \oplus FI(X_L \oplus KO_1, KI_1) \oplus KO_3, KI_3) \quad (4)$$

Equation (4) can be rewritten as

$$FI^{-1}(Y_L \oplus Y_R, KI_3) \oplus KO_3 = X_R \oplus FI(X_L \oplus KO_1, KI_1) \quad (5)$$

Finally, we have

$$X_R \oplus FI(X_L \oplus KO_1, KI_1) = Y_L \oplus FI(X_R \oplus KO_2, KI_2) = FI^{-1}(Y_L \oplus Y_R, KI_3) \oplus KO_3 \quad (6)$$

When input and output of the FO function (X ,Y) are known, then there are 2^{64} keys K such that FO(X,K)=Y.

For arbitrary values of $KI_1, KO_1, KI_2, KI_3 \in \{0,1\}^{16}$, we can efficiently compute KO_2, KO_3 uniquely s.t. $FO(x,K) = y$. If it is needed, we can guess other parts of key and compute remain parts.

D. FL function properties

FL function $FL: \{0,1\}^{32} \times \{0,1\}^{32} \to \{0,1\}^{32}$ is the simplest component of KASUMI. We introduce some properties of FL function.

1. By fixing FL sub-keys, the function is one-to-one w.r.t to the input. But this isn't correct for the sub-keys by fixing the input.
2. If $X = (X_L, X_R)$, $Y = (Y_L, Y_R)$ are input, output of the FL function, and $KL = (KL_1, KL_2)$ is the corresponding sub-key, then

$$(Y_L \oplus X_L)^{>>>1} = Y_R \vee KL_2 \quad (7)$$

$$(Y_R \oplus X_R)^{>>>1} = X_L \wedge KL_1$$

Using (7), it is possible to classify all the FL sub-keys.

In the following sections, we apply our method on reduced rounds KASUMI.

IV. An Algorithm for Generating Equivalence Keys in the First Round

In this section, using mentioned properties for function that used in KASUMI, we propose an efficient algorithm for finding class of keys which map fixed input P_0 to fixed output C_0 for $P_0, C_0 \in \{0,1\}^{64}$ (C_0 is the output of the first round before substitution of two halves)

```
Algorithm 2:

Given $P_0 = (P_L, P_R), C_0 = (C_L, C_R)$
Find every $K \in \{0,1\}^{128}$ s.t.
    $KASUMI_{OneRound}(P_0, K) = C_0$ .

    For $(KL_1, KL_2, KO_1, KI_1, KI_2, KI_3) \in \{0,1\}^{96}$ do
        { find $KO_2, KO_3 \in \{0,1\}^{16}$ s.t.
            $P_R \oplus FO(FL(P_L, KL), KI, KO) = C_R$ }
```

Using FO function property 2, we will be able to look up efficiently desired key values.

$$KO = (KO_1, KO_2, KO_3), \; KI = (KI_1, KI_2, KI_3).$$

There are 2^{96} different keys K that $KASUMI_{FirstRound}(P_0, K) = C_0$ for fixed $P_0, C_0 \in \{0,1\}^{64}$. The sub-keys values $(KO_2, KO_3) \in \{0,1\}^{32}$ can be computed efficiently for every guess of sub-keys $(KL_1, KL_2, KO_1, KI_1, KI_2, KI_3) \in \{0,1\}^{96}$. Therefore current algorithm is efficient, since the complexity of algorithm is 2^{96} and this is the best complexity with regard to the number of keys.

V. An Algorithm for Generating Equivalence Keys in the Second Round

We extend our algorithm to 2-rounds KASUMI. It is expected that there are 2^{64} members in $[K]_{P_0}$ s.t.

P_0 maped C_0 (C_0 is the output of the second round before substitution of two halves). Since we are in the initial rounds of KASUMI and the statistical properties of the cipher algorithm are not complete, the number of elements of class $[K]_{P_0}$ may not be similar to that we expected.

According to cipher structure, there are some relations between input and output of 2-round KASUMI block cipher as follow.

$$C_L \oplus P_R = FO(FL(P_L, RKL_1), RKO_1, RKI_1)$$

$$C_R \oplus P_L = FL(FO(C_L, RKO_2, RKI_2), RKL_2)$$

Where RKL_1 is the sub-key of first round for KL. Consequently, it needs to solve a system of equations for recovering unknown sub-keys.

Based on key schedule, if K_1 is known, the sub-key KL_1 of the first round and KI_3 of the second round, etc, will be known, where $K = (K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_8)$ is the master key of the cipher.

In our algorithm, we guess FL function sub-keys for the first two round, then using FO function equations (3-6), the other sub-keys can be efficiently computed. In fact, in the first two rounds, these FL sub-keys are K_1, K_2, K_3, K_4 . In the rest, we represent our proposed algorithm in which the unknown key values show by capital letters.

Algorithm 3:

```
Given $P_0 = (P_L, P_R), C_0 = (C_L, C_R)$
Find every $K \in \{0,1\}^{128}$ s.t.
$KASUMI_{2\ Round}(P_0, K) = C_0$

For $(k_1, k_2, k_3, k_4) \in \{0,1\}^{64}$ do
  {  $a := FL(P_L, RKL_1)$;
     $b := C_L \oplus P_R$ ;
     $c := C_L$ ;
     $d := FL^{-1}(C_R \oplus P_L, RKL_2)$;
     Solve the following equations,

     $a_R \oplus FI(a_L \oplus k_2^{<<5}, K'_5) = b_L \oplus FI(a_R \oplus K_6^{<<8}, k'_4)$
         $= K_7^{<<13} \oplus FI^{-1}(b_L \oplus b_R, K'_8)$        (8)
     $c_R \oplus FI(c_L \oplus k_3^{<<5}, K'_6) = d_L \oplus FI(c_R \oplus K_7^{<<8}, K'_5)$
         $= K_8^{<<13} \oplus FI^{-1}(d_L \oplus d_R, k'_1)$    }
```

It can be shown that if we would be able to find some part of key, the other parts of key can be computed uniquely. In this algorithm, when we find K_6 , sub-key K_5 can be computed from the first equation of algorithm 2. Using the second equation of algorithm 2, K_7, K_8 can be computed.

For correctness, these sub-key values should be checked by additional check equation in the second part of the first equation of algorithm 2.

Solving strategy in algorithm 2, is an arbitrary strategy that can solve system of equations (8). For example we can replace this strategy with exhaustive search or algebraic methods to find K_6 and then compute the rest of sub-key values.

Complexity of the algorithm 2 depends on solving strategy. In the worst case, if exhaustive search method is used, the complexity of algorithm will be 2^{80} .

VI. EXPERIMENTAL RESULTS

Since KASUMI has 64-bit block length and 128-bit key length, based on birthday paradox, when a fixed plaintext encrypts with 2^{32} different keys, there will be at least two equal cipher-text with probability more than ½. With P=0 (plain-text with 64 bit zero), there are at least two keys that encryption of P lead to same cipher-text.

Using such keys, we get output of the sixth round KASUMI; we wrote equations for last two round and the results are compared. We run our algorithm on 4,6,7-rounds KASUMI separately and obtain some equivalence keys. For 6-round KASUMI, we found three keys with same cipher-text. It can be inferred that 6-round KASUMI shows more non-randomness properties and it can be vulnerability for KASUMI, since we expected two keys with same cipher with considering 2^{32} different keys. One of the results presents in appendix1.

VII. CONCLUSIONS

In this respect, we proposed a new attack on block ciphers with key length greater than block length, if it would be possible to efficiently classify the key space of corresponding block cipher. In fact, the complexity of attack will be $O(max\{2^n, 2^{k-n}\})$ where n is block length and k is key length.

Some recommendations for continuing this work are as follow.

1. Extend our algorithm to higher rounds of KASUMI.
2. Solving algebraic equations obtained by 2-rounds KASUMI in an efficient way.
3. Compute $p(C_i = C'_i \mid C_j = C'_j)$ for $i > j$ where C_i, C_j are the output of i'th-round KASUMI with same plain-text and different keys.

4. Cryptanalysis of 4-rounds KASUMI block cipher with complexity of order 2^{64}.
5. Classification of the key space of other cipher algorithms such as IDEA, 3DES, AES-256 with key length greater than block length.
6. Combining this method with other known attack to achieve a near practical attack algorithm.
7. When there are more than $\left\lceil \frac{k}{n} \right\rceil$ data, our method is not more efficient than the case that we have $\left\lceil \frac{k}{n} \right\rceil$ data. Is it possible to improve this method if we have more data?

References

[1] Matsui, M.: Block Encryption Algorithm MISTY. In: Biham, E. (ed.) FSE 1997. LNCS, vol. 1267, pp. 64–74.Springer, Heidelberg (1997)

[2] Kuhn, U.: Cryptanalysis of Reduced-Round MISTY. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol.2045, pp. 325–339. Springer, Heidelberg (2001)

[3] Jia, K., Li,L., Rechberger, C., Chen, C., Wang, X.: Improved Cryptanalysis of the Block Cipher KASUMI. In:Knudsen, L.R., Wu, H. (eds.) SAC 2012. LNCS, vol. 7707, pp. 222–233. Springer, Heidelberg (2012)

[4] Sugio, N., Aono, H., Hongo, S., Kaneko, T.: A Study on Higher Order Differential Attack of KASUMI. IEICE Transactions 90-A(1), pp. 14-21 (2007)

[5] Blunden, M., Escott, A.: Related Key Attacks on Reduced Round KASUMI. In: Matsui, M. (ed.) FSE 2001.LNCS, vol. 2355, pp. 277–285. Springer, Heidelberg (2002)

[6] Biham, E., Dunkelman, O., Keller, N.: A Related-Key Rectangle Attack on the Full KASUMI. In: Roy, B. (ed.) ASIACRYPT 2005. LNCS, vol. 3788, pp. 443–461. Springer, Heidelberg (2005)

[7] Dunkelman, O., Keller, N., Shamir, A.: A Practical-Time Related-Key Attack on the KASUMI Cryptosystem Used in GSM and 3G Telephony. In: Rabin, T. (ed.) CRYPTO 2010. LNCS 6223, pp. 393–410. Springer,Heidelberg (2010)

[8] K.Jia , L.Li , C.Rechberger, J. Chen , X.Wang, Keting Jia, Christian Rechberger, Xiaoyun Wang, Green Cryptanalysis: Meet-in-the-Middle Key-Recovery for the Full KASUMI Cipher , 2012.

[9] Blog.2ddream.ir/classattack.html, 2013 (in Farsi).

Appendix 1.

```
                Plain-text=0
Key=0xF1D941159CA8B6238135DACB8A370940
      Cipher-text=0x2DBCDA8D84CDAD86

--->> c1: left=0, right=db16eed5
--->> c2: left=db16eed5, right=48d17eb6
--->> c3: left=48d17eb6, right=2ebddad4
--->> c4: left=2ebddad4, right=7b006cf8
--->> c5: left=7b006cf8, right=d8805ffd
--->> c6: left=d8805ffd, right=9f570e58
--->> c7: left=9f570e58, right=84cdad86
--->> c8: left=84cdad86, right=2dbcda8d
```

```
                Plain-text=0

      Key=
0xCAFF6AC383136437A70C4560AC98CE9F
      Cipher-text= 0x2DBCDA8D84CDAD86

--->> c1: left=0, right=aa108129
--->> c2: left=aa108129, right=ec2e85a9
--->> c3: left=ec2e85a9, right=309e5e7b
--->> c4: left=309e5e7b, right=8f1313fb
--->> c5: left=8f1313fb, right=2b23dcc6
--->> c6: left=2b23dcc6, right=9b7de2ee
--->> c7: left=9b7de2ee, right=84cdad86
--->> c8: left=84cdad86, right=2dbcda8d
```

Where “ci” is the output of round i in the KASUMI.