Weak Keys of the Full MISTY1 Recovered in Practical Time

Bungo Taga[1(✉)], Norimitsu Ito[2], and Takako Okano[1]

[1] National Police Academy, 3-12-1, Asahi-cho, Fuchu-shi, Tokyo 183-8558, Japan
{b.taga.54,t.okano}@nparc.npa.go.jp
[2] National Police Agency, 2-1-2, Kasumigaseki, Chiyoda-ku, Tokyo 100-8974, Japan
n.itou.6r.vp@npa.go.jp
https://www.npa.go.jp/keidai/, https://www.npa.go.jp

Abstract. The MISTY1 is a 64-bit block cipher designed by Matsui in 1997. It is listed on the Japanese CRYPTREC Candidate Recommended Ciphers List. Cryptanalysis against the full MISTY1 has already been known, which is the analysis of weak keys in a related-key setting and the integral attack using the division property in a single-key setting. However, these attacks require large amounts of data and time complexity that are practically infeasible. In this paper, we show the existence of new weak keys for the full MISTY1. The MISTY1 can be distinguished from a random permutation and the keys are recovered with a realistically feasible computational complexity, in a related-key setting. It means that a pair of weak keys, one key of which has a specific differential relationship with the other, is used. The computational complexity of the attacks is 2^5 chosen plaintexts for distinguishing the MISTY1 from a random permutation, and 2^8 chosen plaintexts, 2^{25} bytes of memory and a few seconds computed by a desktop PC for key recovery.

Keywords: MISTY1 · Weak keys · Related-key attack

1 Introduction

The MISTY1 [1] is a symmetric key 64-bit block cipher designed by Matsui in 1997, which is listed on the Japanese CRYPTREC Candidate Recommended Ciphers List [2] and has been standardized in NESSIE [4], ISO/IEC [5] and RFCs [6].

Theoretical attacks on the full-round MISTY1 are already known. We summarize attacks on the full-round MISTY1 in Table 1. In 2013, Lu et al. showed the existence of weak keys in a related-key setting [7], followed by the related work [8]. In 2015, Todo presented the integral attack in a single-key setting using the division property [9], followed by the related work [10,11]. However, these attacks are not yet a realistic threat due to the very large amount of computational complexity (Table 1).

K. Minematsu and M. Mimura (Eds.): IWSEC 2024, LNCS 14977, pp. 65–81, 2024.
https://doi.org/10.1007/978-981-97-7737-2_4

Table 1. Attacks on the full MISTY1

Attack	Keys	Data	Time	Memory
Related-key differential [7]	$2^{103.57}$	2^{61}CC	$2^{90.93}$	$2^{99.2}$Bytes
Related-key amplified boomerang [8]	2^{92}	$2^{60.5}$CP	$2^{87.33}$	$2^{80.07}$Bytes
Related-key differential				
Distinguisher Section 4.1	$\mathbf{3 \cdot 2^{74}}$	$\mathbf{2^5}$**CP**	$\mathbf{2^9}^{\dagger}$	
Key recovery Section 4.2	$\mathbf{3 \cdot 2^{74}}$	$\mathbf{2^8}$**CP**	$\lesssim \mathbf{2^{29}}$	$\mathbf{2^{25}}$**Bytes**
Integral [9]	2^{128}	$2^{63.58}$CP	2^{121}	not specified
Integral [10]	2^{128}	$2^{63.994}$CP	$2^{108.3}$	not specified
Integral [11]	2^{128}	$2^{63.9999}$CC	2^{79}	not specified
Integral [11]	2^{128}	$2^{64} - 2^{36}$CPC	$2^{69.5}$	not specified

CP: chosen plaintexts, CC: chosen ciphertexts
CPC: chosen plaintexts and ciphertexts
† The unit of time is the time for comparing two ciphertexts
In the other cases it is the time for encrypting one time.

Weak keys mean the keys whose use would cause some kind of unexpected behavior in this paper. The related-key attacks [14] are attacks under the condition that a ciphertext and the corresponding plaintext encrypted with multiple keys that are related to each other are available.

Our contributions presented in this paper are as follows.

- We found weak keys of the MISTY1, which have not been previously shown.
- We showed that it is possible to distinguish the MISTY1 from random permutations and to recover the keys when the weak keys are used.
- We estimated the computational complexity required for these attacks, and by conducting computer experiments using a desktop PC we demonstrated that the keys are recovered in less than a few seconds.

Note that the attacks are not considered to be a realistic threat for two reasons. Firstly, they are related-key attacks where two weak keys with a differential relationship between them convenient for attackers need to be used. Secondly, the number of weak keys we found is $3 \cdot 2^{74}$. It is not small, but extremely smaller than the total number of keys.

This paper is organized as follows. In Sect. 2 we describe the notation used in this paper, weak keys, related-key attacks and the structure of the MISTY1. In Sect. 3 we analyze the key scheduling part of the MISTY1 to derive the weak keys, and show that there is a differential characteristic in the data randomizing part with a very large differential probability in a related-key setting. In Sect. 4, we construct a related-key distinguisher of the MISTY1 and recover secret key exploiting the differential characteristic shown in Sect. 3. In Sect. 5 we summarize our results.

2 Preliminaries

In this section we describe the notation used in this paper, weak keys, related-key attacks and the structures of the data randomizing and key scheduling parts of the MISTY1.

2.1 Notation

The notation used throughout this paper is as shown in the reference [3] and Table 2.

Table 2. The notation used in this paper

Subscript: $_{\text{in/out}}$	Denotes the input/output data of the function e.g., FO_{in}, FO_{out}
Subscript: $_{i}$	Denotes the i-th 16-bit data from the left
Subscript: $_{(\text{R}j)}$	The right (lower) j-bit value of 16-bit data
Subscript: $_{(\text{L}j)}$	The left (upper) j-bit value of 16-bit data
e.g.,	$P = P_1 \| P_2 \| P_3 \| P_4$ $K_1 = K_{1(\text{L9})} \| K_{1(\text{R7})} = K_{1(\text{L7})} \| K_{1(\text{R9})}$
Prefix: Δ	Denotes a differential between two data
`Typewriter style`	Hexadecimal notation (e.g., `0a`)
$\Delta K := K_{\text{sec}} \oplus K_{\text{rel}}$	The differential between the secret key and the related key
$\Delta K'$	The differential between the two corresponding extended keys

2.2 MISTY1

The MISTY1 is a symmetric key block cipher with a data block length of 64 bits and a secret key length of 128 bits. It consists of two parts: the data randomizing part, which randomizes a 64-bit plaintext and outputs the 64-bit ciphertext, and the key scheduling part, which outputs the 128-bit extended key for an input of 128-bit secret key. Both the secret key and the extended key are used in the data randomizing part. The structures of the data randomizing and the key scheduling parts are explained below.

2.2.1 The Data Randomizing Part The structure of the data randomizing part of the MISTY1 is shown in Fig. 1. It is a Feistel structure in which the input 64-bit plaintext is divided into two 32-bit blocks, and each block is transformed alternately by the function FO. The function FO and its internal function, the function FI, also have Feistel structures. As a whole, the data randomizing part has a three-layer nested structure, and also has a structure in which the transformation is repeated with two function FLs and two function FOs as a unit.

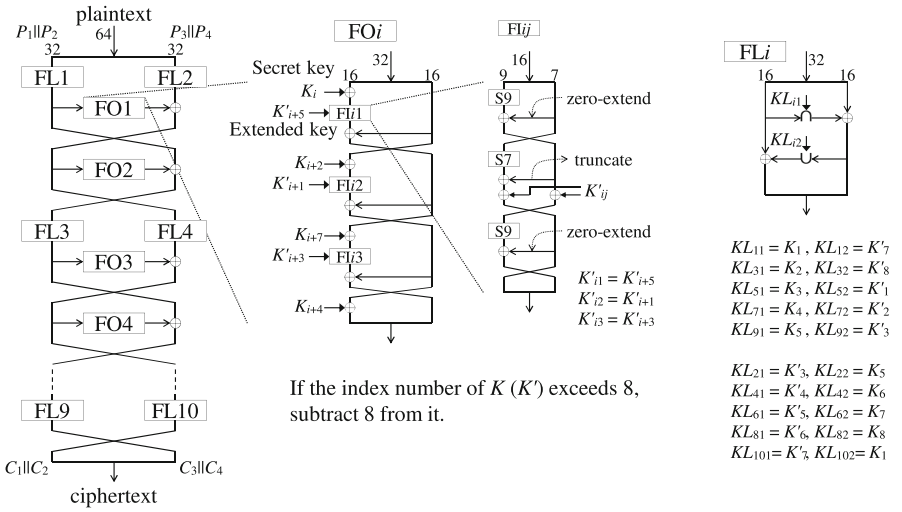

Fig. 1. The Data randomizing part of the MISTY1

The keys used in the data randomizing part are the 16-bit secret keys K_1 to K_8 and the extended keys K'_1 to K'_8, which are described in Sect. 2.2.2. In the function FO, the extended keys are used in its internal functions, the function FIs, and the secret keys are used outside the function FIs. In the function FL, one secret key and one extended key are used.

The non-linear operations with respect to the exclusive-or (XOR) operation are the bitwise AND ($\cap$) and OR ($\cup$) operations in the function FL and are the substitution tables S7 and S9 in the function FI. Within one function FO, the transformations by substitution tables S7 and S9 are performed many times, which is the reason why the differential probability in single-key settings is small. If there are differential characteristics of the function FO with a large differential probability, the differential probability of the entire data randomizing part may be also large.

2.2.2 The Key Scheduling Part The structure of the key scheduling part of the MISTY1 is shown in Fig. 2. In the key scheduling part, the 128-bit secret key is divided into eight 16-bit secret keys K_1 to K_8, each 16-bit key is input to two adjacent function FIs and eight 16-bit extended keys K'_1 to K'_8 are output. Each extended key K'_i is obtained as an output for two secret key inputs, K_i and K_{i+1}, to the function FI. In the function FI, the substitution tables S9 and S7 are used, which are non-linear operations with respect to the XOR operation. The function FI is identical to that in the data randomizing part.

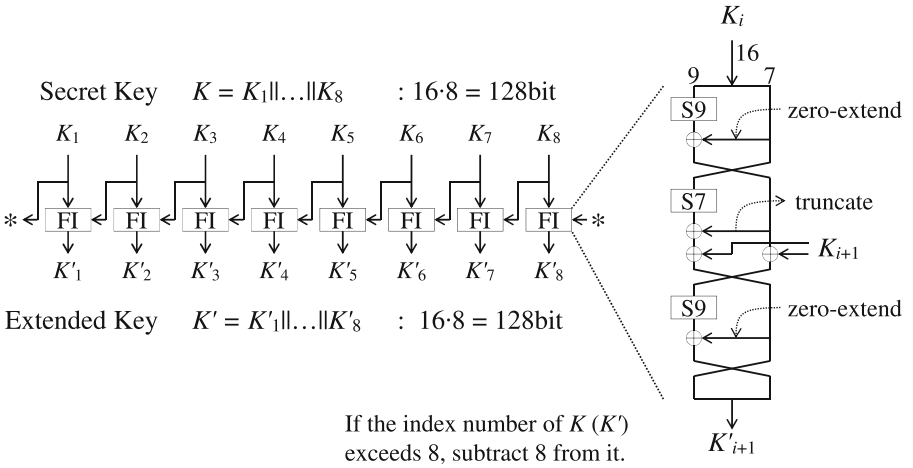

Fig. 2. The Key Scheduling part of the MISTY1

2.3 Weak Keys

In this paper, weak keys are defined as the keys whose use would cause some kind of unexpected behavior as mentioned in Sect. 1. Although, a weak key was initially defined as a key that reverts to the original plaintext in the block cipher DES [12] when encrypted twice with the same key for any plaintext, the former definition is used in the previous works [7,8]. When a pair of weak keys shown in this paper is used, we can distinguish MISTY1 from a random permutation and recover the keys.

2.4 Related-Key Attacks

Related-key attacks were proposed in the early 1990 s [14,15]. Biryukov et al. illustrated the first attack on the full-round AES-192 and AES-256 [13] in a related-key setting, which distinguished AES from a random permutation and recovered an AES encryption key more efficiently than a brute force attack [16, 17].

In a related-key setting, attackers can obtain pairs of plaintexts and the corresponding pairs of ciphertexts associated with each other by related keys besides the encryption key. Attackers can also know and control the relationships between the encryption key and related keys, even if they do not know the encryption key itself. Thus, the related-key attack is an attack under very favourable conditions for the attackers, and conversely, the conditions for the attack to be successful are so severe that it is considered to be an attack with a small chance of being realized in normal encryption applications. However, depending on the method of secret key generation and distribution, etc., an attacker may be able to obtain the differential between multiple secret keys, and therefore the related-key attack may be a realistic attack.

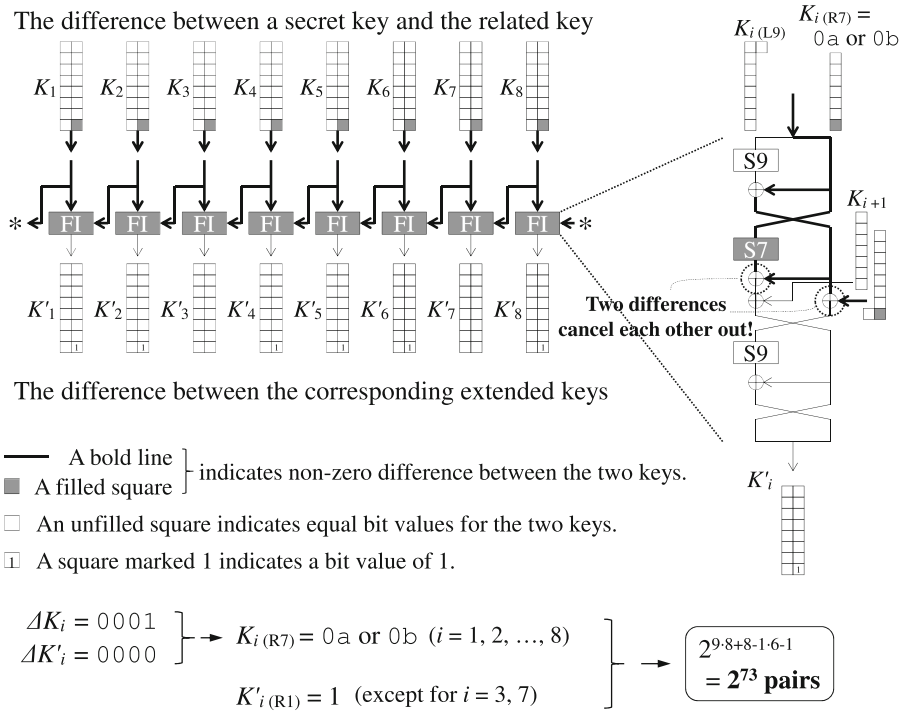

Fig. 3. Differences between weak key pairs

In this paper, we consider the related-key differential attack shown below:

$$\begin{aligned} P &\xrightarrow{K} C \\ P \oplus \Delta P &\xrightarrow{K \oplus \Delta K} C \oplus \Delta C, \end{aligned}$$

where P, C and K denote a plaintext, a ciphertext and a secret key and ΔP, ΔC and ΔK are differences between the two plaintexts, the two ciphertexts and the two secret keys, respectively. When Ps, ΔP, and ΔK are chosen to be convenient to the attacker, the probability of differential characteristic is large enough for the attacker to distinguish MISTY1 from a random permutation and to recover the secret key.

3 Weak Keys of the MISTY1

When the weak key shown in this paper is used together with another weak key with a specific differential relationship as the related key, the MISTY1 can be distinguished from a random permutation and the secret key can be recovered. In this section, we first illustrate what pairs of secret keys and related keys

can be weak keys, and then show that when these pairs of weak keys are used, there exists a differential characteristic with very large probability in the data randomizing part.

3.1 Pairs of Weak Keys and the Differentials in Them

Table 3. Tuples of differences between two weak keys, and the lower 7-bit values of the weak keys

h	Tuples of $\Delta K_{i(\mathrm{R7})}$ and $(K_{\mathrm{sec}i(\mathrm{R7})}, K_{\mathrm{rel}i(\mathrm{R7})})$	Number of keys
1	01(0a,0b),20(0c,2c),40(18,58)	$\mathbf{3 \cdot 2^{74}}$
2	03(61,62),06(31,37),09(53,5a),0c(30,3c),18(4c,54), 21(45,64),24(5e,7a),28(50,78),30(41,71),41(26,67), 42(35,77),60(2e,4e)	$12 \cdot 2^{68}$
3	07(72,75),0d(21,2c),13(28,3b),1c(2e,32),25(48,6d), 26(19,3f),29(1c,35),38(01,39),43(03,40),46(07,41), 49(08,41),52(0b,59),61(1e,7f),64(36,52),68(06,6e), 70(0d,7d)	$16 \cdot 2^{62}$
4	0f(07,08),1b(2f,34),1d(25,38),1e(45,5b),27(15,32), 2d(0c,21),2e(46,68),3c(1e,22),4b(02,49),4d(39,74), 53(0a,59),71(1a,6b),78(27,5f)	$13 \cdot 2^{56}$
5	1f(00,1f),2f(04,2b),3b(15,2e),3e(58,66),57(09,5e), 5b(15,4e),5d(22,7f),6b(1c,77),73(09,7a),75(01,74), 76(07,71),79(08,71),7c(32,4e)	$13 \cdot 2^{50}$
6	3f(5b,64),7d(2a,57),7e(18,66)	$3 \cdot 2^{44}$
7	N/A	

h: Hamming weight of $\Delta K_{i(\mathrm{R7})}$

The key scheduling part is a function that outputs a 128-bit extended key for an input of 128-bit secret key (Fig. 2). As this function is not bijection, there may be more than one secret key such that the key scheduling part outputs extended keys with the same value. For example, if the conditions $K_{\mathrm{sec}i(\mathrm{R7})} =$ 0a or 0b and $\Delta K_i := K_{\mathrm{sec}i} \oplus K_{\mathrm{rel}i} =$ 01 $(i = 1, \cdots, 8)$ for the secret key and the related key pair $(K_{\mathrm{sec}}, K_{\mathrm{rel}})$ are satisfied, the key scheduling part outputs the extended keys with the same value.

This weak key pair thus satisfies the condition that the lower 7-bit value of a 16-bit secret key, $K_{\mathrm{sec}i(\mathrm{R7})}$, is one of the two values and that the differential between K_{sec} and K_{rel}, ΔK_i, is the same for all i. Furthermore, another condition is imposed on the extended key K' as described in Sect. 3.2.2. The conditions to be satisfied by the weak key pair are as follows:

Conditions on pairs of weak keys

$$\Delta K_{i(\mathrm{L9})} = 0 \ (i = 1, \cdots, 8), \quad \Delta K_{1(\mathrm{R7})} = \Delta K_{2(\mathrm{R7})} = \cdots = \Delta K_{8(\mathrm{R7})} \tag{1}$$

$$\Delta K'_i = 0 \ (i = 1, \cdots, 8) \tag{2}$$

$$K'_{i(\mathrm{R7})} \cap \Delta K_{i(\mathrm{R7})} = \Delta K_{i(\mathrm{R7})} \ (\text{except for } i = 3, 7) \tag{3}$$

Figure 3 shows a differential between a pair of weak keys in case of $\Delta K_{i(\mathrm{R7})} =$ `01`. To satisfy the condition (2), the input and output differential values of S7 must be equal in the differential path on the right-hand side of Fig. 3, and such tuples of input differentials $\Delta K_{i(\mathrm{R7})}$ and pairs of input values ($K_{\mathrm{sec}i(\mathrm{R7})}$, $K_{\mathrm{rel}i(\mathrm{R7})}$) for S7 are limited to the 60 tuples shown in Table 3. $\Delta K_{i+1(\mathrm{R9})}$ on the right-hand side of Fig. 3 always cancels out with the input differential $\Delta K_{i(\mathrm{R9})}$ according to the condition (1).

The condition (3) is imposed on the extended key that the bit value of the extended key K'_i corresponding to the bit position with a non-zero difference of ΔK_i is 1. This is necessary for the differential characteristic of the function FL described in Sect. 3.2.2 to be possible.

The number of weak keys is estimated as follows. From the conditions (1) and (2), each $K_{\mathrm{sec}i(\mathrm{L9})}$ can take 2^9 values and each $K_{\mathrm{sec}i(\mathrm{R7})}$ can take two values. If the Hamming weight of $\Delta K_{i(\mathrm{R7})}$ is h, from condition (3) h bits of the extended key K'_i must be 1 (except for $i = 3, 7$). Therefore, for each $\Delta K_{i(\mathrm{R7})}$ in Table 3, the number of weak keys is $2^{9\cdot 8+8-6h} = 2^{80-6h}$ and decreases exponentially as h increases.

The number of weak keys is the largest when $\Delta K_{i(\mathrm{R7})} =$ `01, 20, 40` with $h = 1$, which is $2^{9\cdot 8+8-6\cdot 1} = 2^{74}$ each. It is also when h=1 that the differential probability of the data randomizing part is the largest, as described in Sect. 4. In the following, unless otherwise stated, we will discuss the case $\Delta K_{i(\mathrm{R7})} =$ `01`, but exactly the same argument holds for the cases $\Delta K_{i(\mathrm{R7})} =$ `20` and `40`.

3.2 Differential Characteristics

It has been shown by the developers that the MISTY1 is based on the theory of provable security against differential cryptanalysis in single-key settings [1]. When the differential path is actually explored, it can be seen that the differential probability is significantly small in the function FOs. If the differential probability is sufficiently large in each function FO, the differential probability of the entire data randomizing part may be large.

In this section, we first show that there exists a differential characteristic for the function FO with a differential probability of 1 when a pair of weak keys satisfying conditions (1) to (3) in Sect. 3.1 is used. Next, it is shown that for the entire data randomizing part, there exists a differential characteristic where the differential probability of all function FOs is 1, and the differential characteristics and differential probabilities of the function FLs in that case are shown.

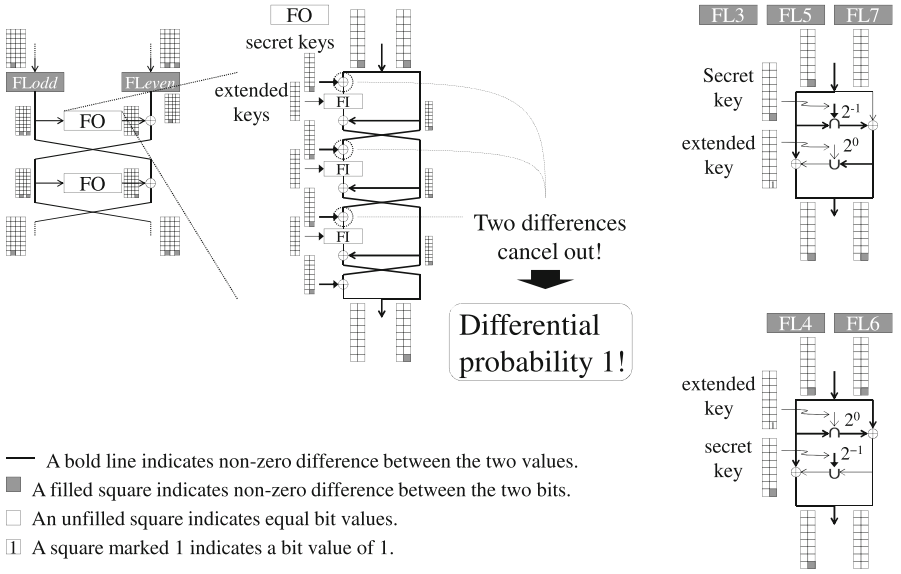

Fig. 4. Differential characteristics of the 2-round function that can be exploited for the attacks

3.2.1 The Function FO For the case $\Delta K_{i(\mathrm{R7})} = \mathtt{01}$, the differential characteristic of the function FO, where the differential probability is 1, is shown in the center of Fig. 4. In this differential characteristic, the input differential values of the function FI are always zero because a difference between two secret keys, ΔK_i, and the difference between two input values cancel just before the function FI. Also, since $\Delta K_i' = 0$ for the extended key, no differences occur in the function FI. This results in an output differential value $\Delta \mathrm{FO}_{\mathrm{out}} = \mathtt{0000\ 0001}$ with probability 1 for the input differential value $\Delta \mathrm{FO}_{\mathrm{in}} = \mathtt{0001\ 0001}$ of the function FO.

Note that such differential paths of the function FO exist that the differential probability be 1 not only when the Hamming weight of $\Delta K_{i(\mathrm{R7})}$ is 1, but also for all 60 cases in Table 3.

3.2.2 The Function FL The data randomizing part has a structure in which transformations using two function FLs and two function FOs are repeated as a unit. So, if a differential characteristic exists in which the input differential value and output differential value are equal and the differential probability is sufficiently large in the transformation of this unit, then a differential characteristic with a large differential probability also exists for the entire data randomizing part.

Using the differential characteristic of the function FO with probability 1 described in Sect. 3.2.1 to search for the differential characteristics in the transformation of the unit iteration described above, it can be seen that in order for

the input and output differential values to be equal, the input and output differences of the odd-th (FL*odd*) and even-th (FL*even*) of the function FL must be the following values (right-hand of Fig. 4).

$$\begin{aligned} \text{Odd-th}: \Delta\text{FL}odd_{\text{in}} &= \texttt{0001 0000},\ \Delta\text{FL}odd_{\text{out}} = \texttt{0001 0001} \\ \text{Even-th}: \Delta\text{FL}even_{\text{in}} &= \texttt{0001 0001},\ \Delta\text{FL}even_{\text{out}} = \texttt{0001 0000} \end{aligned} \tag{4}$$

In order for the differential property of the function FL taking such input and output differences to actually exist, condition (3) of Sect. 3.1 is required for the extended key K'_i, as follows.

In the FL*odd*, the condition (3) of Sect. 3.1, where the least significant bit of the extended key K'_i is 1, is required because the output difference of the $\cup$ operation must be zero. Conversely, if this condition is satisfied, the differential after the $\cup$ operation is 0 with probability 1. The differential probability is 2^{-1} in the $\cap$ operation and therefore also 2^{-1} for the whole FL*odd* (top right of Fig. 4).

Since the output difference of the $\cap$ operation must be 0001 in the FL*even*,, the condition that the least significant bit of the extended key K'_i is 1 is required, as in the FL*odd* case. If this condition is satisfied, the difference of the $\cap$ operation is 0001 with probability 1. The differential probability is 2^{-1} in the $\cup$ operation and therefore also 2^{-1} for the whole FL*even* (bottom right of Fig. 4).

In general, the differential probability of the function FL when the Hamming weight of $\Delta K_{i(\texttt{R7})}$ is h is found to be 2^{-h} by the same argument. In addition, the condition of a bit value of 1 for h bits of the extended key K'_i is required, which means that the number of weak keys decreases exponentially as h increases (Table 3).

3.2.3 The Data Randomizing Part Figure 6 shows the differential characteristics used in the attacks. The left-hand side is used to distinguish the MISTY1 from a random permutation, while the right-hand side is used for key recovery.

In the function FL1 and FL2, each of the upper 15 bits of the plaintext difference ΔP_1 to ΔP_4 must be 0 and each of the least significant bit of them are not restricted, while the output difference must satisfy the condition (4) in Sect. 3.2.2. In the function FL9 and FL10, even if the output differential value does not satisfy condition (4), the attack is possible because the upper 15 bits of the ciphertext differentials ΔC_1 and ΔC_2 are 0. Therefore, condition (3) in Sect. 3.1 is not necessary for the extended keys K'_3 and K'_7, which are used in the function FL1, 2, 9 and 10 but not in the function FL3 to 8. The function FL8 is discussed later.

In the attacks, plaintext pairs that take the differential values of all patterns with respect to the least significant bits of ΔP_1 to ΔP_4 are used.

The differential values of the ciphertext used in the attack differ between the case of distinguishing from a random permutation and the case of key recovery. For ΔC_1 and ΔC_2, the upper 15-bit values are 0 in both cases, whereas for ΔC_3 and ΔC_4, there is no restriction in the case of distinguishing from a random permutation, but the upper 15-bit value is non-zero for key recovery.

The differential probabilities are as follows. The differential probabilities for the function FL3 to FL7 are 2^{-1} each, as described in Sect. 3.2.2. For the function FL1 and FL2, the differential probability is 2^{-1} for each of the $\cap$ and $\cup$ operations, so the probabilities of the function FL1 and FL2 are both 2^{-2}. For the function FL8, the differential probabilities are different in the case of distinguishing from a random permutation and the case of key recovery. For the input difference $\Delta\mathrm{FL8}_{\mathrm{in}}$ = `0001 0001`, the output difference is either $\Delta\mathrm{FL8}_{\mathrm{out}}$ = `0001 0000` or `0000 0000`, both with a differential probability of 2^{-1}. In the case of distinguishing from a random permutation, the attack succeeds because $\Delta C_{1(\mathrm{L15})} = \Delta C_{2(\mathrm{L15})} = 0$, regardless of the value of $\Delta\mathrm{FL8}_{\mathrm{out}}$, so the differential probability of the function FL8 can be regarded as 1. On the other hand, in the case of key recovery, the differential probability of the function FL8 is 2^{-1} because $\Delta\mathrm{FL8}_{\mathrm{out}}$ must be = `0000 0000`. For the function FL9 and FL10, the differential probability can be regarded as 1, as there is no restriction imposed on the output differences. Therefore, the differential probability for the entire data randomizing part is $2^{-(2\cdot 2+1\cdot 5)} = 2^{-9}$ for distinguishing from a random permutation and $2^{-(2\cdot 2+1\cdot 6)} = 2^{-10}$ for key recovery.

4 Attacks on the MISTY1

In this section, we show the attack procedure for distinguishing the MISTY1 from a random permutation and for key recovery by using the differential characteristics of the data randomizing part described in Sect. 3.2.

4.1 Distinguisher

In this attack, plaintext pairs need to be encrypted with a weak key and the related key, respectively, and the corresponding ciphertext pairs be compared with each other, in order to distinguish MISTY1 from a random permutation. The values of the chosen plaintext and the corresponding ciphertext are assumed to be known to the attacker. The attack procedure and computational complexity are shown later.

The Attack Procedure. The attacker chooses plaintexts according to the following procedure.

(i) For an arbitrarily chosen plaintext $P = P_1||P_2||P_3||P_4$, fix the upper 15 bits of each P_i, take the all values for the least significant bit, and a total of $2^4 = 16$ plaintexts make up a set.
(ii) Choose a total of 2^n sets of plaintexts described in (i) that differ from each other (the total number of chosen plaintexts is $2^n \cdot 2^4 = 2^{n+4}$).

Next, the attacker obtains two ciphertexts encrypted with the weak key and the related key, respectively, for each of the $2^4 = 16$ plaintexts, constructs $2^4 \cdot 2^4 = 2^8$ ciphertext pairs for each set of plaintexts, that is, $2^n \cdot 2^8 = 2^{n+8}$ ciphertext

pairs for 2^n sets of plaintexts. The paired ciphertexts are compared with each other and if a ciphertext pair is found for which $\Delta C_{1(\mathrm{L15})} = \Delta C_{2(\mathrm{L15})} = 0$, the attack is considered successful. $\Delta C_{3(\mathrm{L15})}$ and $\Delta C_{4(\mathrm{L15})}$ may or may not be zero and are determined by the output difference of the function FL8.

Complexity. Since the differential probability of the data randomizing part is 2^{-9} as calculated in Sect. 3.2.3, one ciphertext pair per $2^{n+8} = 2^9$ pairs is expected to satisfy the condition $\Delta C_{1(\mathrm{L15})} = \Delta C_{2(\mathrm{L15})} = 0$. That is, if the attacker obtains about 2^9 ciphertext pairs, then the attacker can distinguish MISTY1 from a random permutation. The data complexity required is $2^{n+4} = 2^5$ chosen plaintexts and the time complexity is the comparison of 2^9 pairs of ciphertexts.

4.2 Key Recovery

Table 4. Examples of pairs of CPs and the corresponding ciphertexts that are exploited for key recovery

Pairs of CPs	ciphertexts
b707 5efb b524 23a5	9e1c 104c eaac 4784
b706 5efb b525 23a5	9e1c 104d b9a7 285f
1622 a031 4495 b33c	06e8 0d03 2b57 5ae9
1623 a030 4495 b33c	06e9 0d02 a0a6 7273
01b9 8a3f 1b4e 5471	2678 717a 8e1a 9671
01b8 8a3f 1b4f 5471	2678 717b 062f 5fbb
170c 43b5 75e6 01e4	aeaa 0b5c dd53 bc03
170d 43b4 75e6 01e4	aeaa 0b5d 8d91 21f8
499f f3d3 916e 382b	a8bb 1931 b66a 9275
499e f3d3 916f 382b	a8ba 1930 a624 0556
2281 76ff 86c0 1446	b072 ab51 4865 11eb
2280 76ff 86c0 1446	b073 ab50 3978 f941
2280 76ff 86c0 1447	2484 25c8 fcef c910
2281 76fe 86c1 1447	2484 25c9 cd4e 2ac5
366f 51ca b97c e559	b7d0 ea3c 941a 0e58
366e 51ca b97d e559	b7d0 ea3d 7830 3fa0
e724 f9e3 960d ce69	dc99 dd98 7143 8869
e725 f9e2 960c ce69	dc99 dd99 e8dd 133e
ΔK_i=0001	
K_{sec} = 170b 438a 758b 018b 498b f38a 910a 380b	
K_{rel} = 170a 438b 758a 018a 498a f38b 910b 380a	

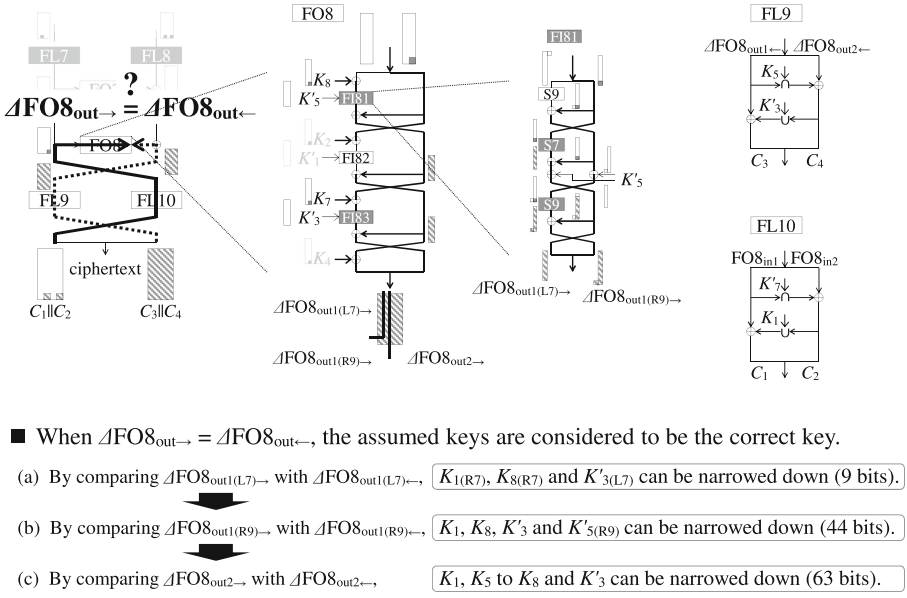

Fig. 5. Key recovery for K_1, K_5 to K_8 and K'_3

In this attack, the attacker recovers the weak key by exploiting the ciphertext pairs obtained by encrypting the chosen plaintexts with the weak key and the related key, respectively. The attacker needs to know the values of the ciphertexts and that the ciphertext pairs are obtained by encrypting the chosen plaintexts with a weak key and the related key, but does not need to know the values of the plaintexts.

The Attack Procedure. The attacker chooses 2^n sets of plaintexts to form 2^{n+8} pairs of ciphertexts by the same procedure as for distinguisher, and picks up the ciphertext pairs with $\Delta C_{1(\mathrm{L15})} = \Delta C_{2(\mathrm{L15})} = 0$, $\Delta C_{3(\mathrm{L15})} \neq 0$, $\Delta C_{4(\mathrm{L15})} \neq 0$ among these pairs. Since the differential probability is 2^{-10} as obtained in Sect. 3.2.3, the expected value of the number of the ciphertext pairs to be picked up is 2^{n-2}.

The attacker then calculates $\Delta\mathrm{FO8}_{\mathrm{out}}$ in two different ways, denoted as $\Delta\mathrm{FO8}_{\mathrm{out}\rightarrow}$ and $\Delta\mathrm{FO8}_{\mathrm{out}\leftarrow}$. The subscript $\rightarrow$ shows that $\Delta\mathrm{FO8}_{\mathrm{out}}$ are obtained by calculating the inverse function of FO10 and the function FO8, and the subscript $\leftarrow$ shows that $\Delta\mathrm{FO8}_{\mathrm{out}}$ are obtained by the inverse function of FO9. $\Delta\mathrm{FO8}_{\mathrm{out}\rightarrow}$ depends on C_1s, C_2s, K_1, K_7, K_8, $K'_{3(\mathrm{R9})}$ and K'_5, and $\Delta\mathrm{FO8}_{\mathrm{out}\leftarrow}$ is obtained depends on C_3s, C_4s, K_5 and K'_3 (see Fig. 5). The correct key candidate is narrowed down using the fact that $\Delta\mathrm{FO8}_{\mathrm{out}\rightarrow}$ is always equal to $\Delta\mathrm{FO8}_{\mathrm{out}\leftarrow}$ if the key candidate is correct and that $\Delta\mathrm{FO8}_{\mathrm{out}\rightarrow} \neq \Delta\mathrm{FO8}_{\mathrm{out}\leftarrow}$ with a high probability if it is incorrect.

The keys that need to be assumed are K_1, K_5 to K_8 and K'_3, as K'_5 is obtained from K_5 and K_6 by calculating the function FI. According to the conditions (1) to (3) of Sect. 3.1, 6 of the lower 7 bits in each K_i and 1 bit in each of the K'_5, K'_6 and K'_8 are known, so the total number of bits of the key to need to be assumed is $5 \cdot 10 - 3 + 16 = 63$ bits.

The search for the entire 63 bits of the key is very time-consuming using a desktop PC. Therefore, $\Delta\mathrm{FO8}_{\mathrm{out}}$ was divided into three blocks to reduce the search time in our analysis, which are (a) $\Delta\mathrm{FO8}_{\mathrm{out1(L7)}}$, (b) $\Delta\mathrm{FO8}_{\mathrm{out1(R9)}}$ and (c) $\Delta\mathrm{FO8}_{\mathrm{out2}}$, and then key recovery was performed by calculating (a) to (c) in sequence.

First, in the calculation of (a), $K_{1(\mathrm{R7})}$, $K_{8(\mathrm{R7})}$ and $K'_{3(\mathrm{L7})}$ are narrowed down by comparing $\Delta\mathrm{FO8}_{\mathrm{out1(L7)}\rightarrow}$ with $\mathrm{FO8}_{\mathrm{out1(L7)}\leftarrow}$. Next, in the calculation of (b), K_1, K_8, K'_3 and $K'_{5(\mathrm{R7})}$ are narrowed down by comparing $\Delta\mathrm{FO8}_{\mathrm{out1(R9)}\rightarrow}$ with $\mathrm{FO8}_{\mathrm{out1(R9)}\leftarrow}$. Then, in the calculation of (c), K_1, K_5 to K_8 and K'_3 are narrowed down by comparing $\Delta\mathrm{FO8}_{\mathrm{out2}\rightarrow}$ with $\mathrm{FO8}_{\mathrm{out2}\leftarrow}$. Finally, an exhaustive search of the remaining unknown bits containing K_2 to K_4 completes key recovery for all bits. Note that 6 bits of each of K_2 to K_4 and the least significant bit of K'_1, K'_2 and K'_4 are known, and the 16 bits of K'_3 have been narrowed down in (a) to (c). The time complexity is reduced because it can be expressed as the sum of those required for each procedure, not the product.

In the case of $\Delta K_{i(\mathrm{R7})} = \mathtt{01}$, 6 of the lower 7 bits of each K_i and the least significant bit each of K'_1, K'_2, K'_4 to K'_6 and K'_8 are already known before these procedures are performed, so $8 \cdot (16-6) - 6 \cdot 1 = 74$ bits of the keys are recovered in these procedures.

As the key assumptions required for the $\Delta\mathrm{FO8}_{\mathrm{out}\rightarrow}$ and $\Delta\mathrm{FO8}_{\mathrm{out}\leftarrow}$ calculations can be made independently, the number of calculations can be reduced in exchange for requiring memory by expanding the data related to the key assumed in one of them into memory.

Table 5. The number of ciphertext pairs and time required for key recovery

The number of ciphertext pairs: N	3	4	5	6
Average timeseconds	54	1.4	0.81	0.71
Time complexity†	$\lesssim 2^{33.7}$	$\lesssim 2^{28.8}$	$\lesssim 2^{27.2}$	$\lesssim 2^{27.0}$
Memory (MiB)	$\lesssim 82$	32	32	32

† The unit of time is the time for encrypting one time
($\simeq$ the time for calculating the function FO8 eight times)

Complexity. In the calculation of (a), 9 bits of the keys are searched exhaustively, which are the 1 bit of $K_{1(\mathrm{R7})}$, the 1 bit of $K_{8(\mathrm{R7})}$ and the 7 bits of $K'_{3(\mathrm{L7})}$. In the calculation of (b), 34 bits of the keys, which are the 17 bits of $K_{1(\mathrm{L9})}$ and $K_{8(\mathrm{L9})}$, the 9 bits of $K'_{3(\mathrm{R9})}$ and the 8 bits of $K'_{5(\mathrm{R9})}$, are searched, in addition to the bits that are narrowed down in (a). Note that the condition (3) on

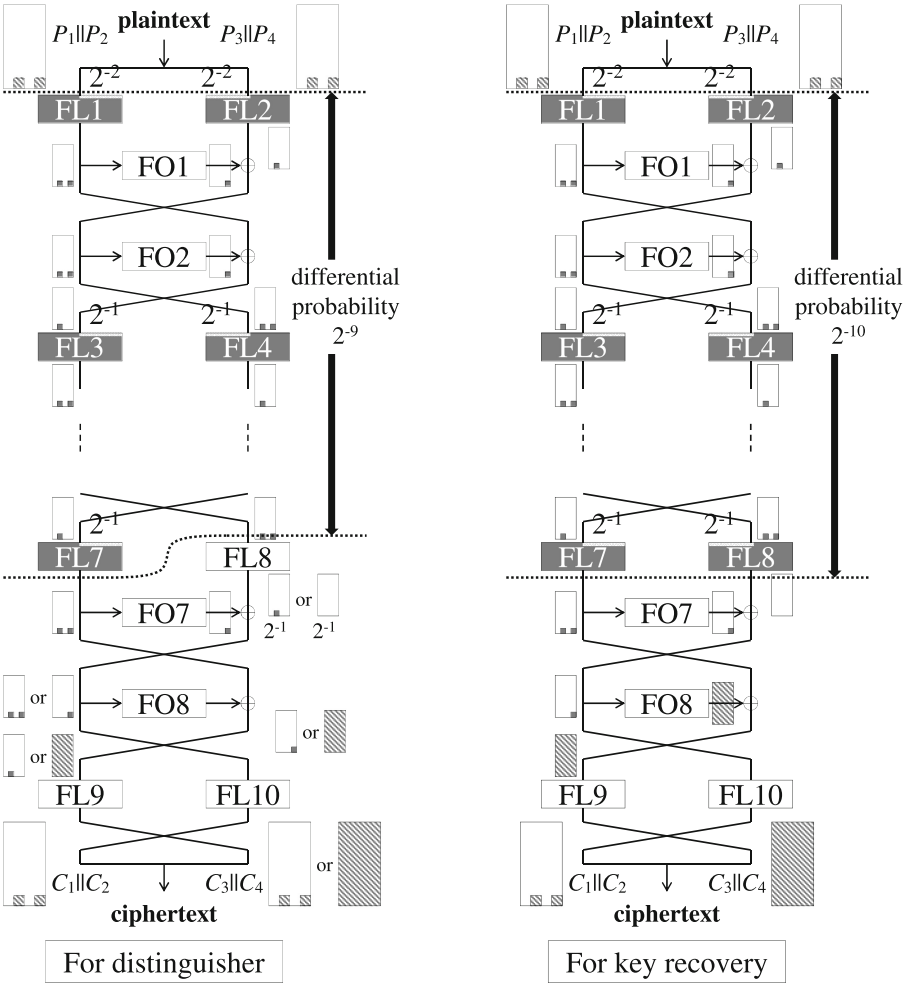

Fig. 6. Differential characteristics of the data randomizing part that can be exploited for the attacks

K'_8 reduces the complexity of searching K_1 and K_8 by 1 bit. Furthermore, by narrowing down K_1, K_8 and $K'_{5(\mathrm{R9})}$ prior to narrowing down $K'_{3(\mathrm{R9})}$, though memory complexity is increased, the time complexity is reduced. In the calculation of (c), $10 \cdot 3 - 9 - 1 = 20$ bits of K_5, K_6 and K_7 in addition to the bits that are narrowed down in (a) and (b) are searched exhaustively. Note that the complexity of searching K_5, K_6 and K_7 is reduced by 9 bits because the $K'_{5(\mathrm{R9})}$

are narrowed down in (b) and by 1 bit because of the condition 3 on K'_6. The key to be searched for in the calculations of (a) to (c) is 63 bits in total, as described above.

If the number of ciphertext pairs used for key recovery is small, then the keys cannot be sufficiently narrowed down in each calculation of (a) to (c) as a result the time complexity increases in subsequent calculations. Also, the time/memory complexity varies depending on the value of the ciphertext pairs. Therefore, it is difficult to estimate the time/memory complexity theoretically. So, we measured the time, the number of calculations of the function FO and amount of memory required for the key recovery by computational experiment. The results are shown in Table 4. The desktop PC used was as follows.

Processor:	Intel(R) Core (TM) i9-9900K CPU
Memory:	32GB
Development environment:	Microsoft Visual C++ 2015

In Table 5, N $(= 2^{n-2})$ is the number of ciphertext pairs used, and Average time is the average of the search time required for all combinations of N pairs from the 9 ciphertext pairs in Table 4. The search time was about 1.4 s when $N = 4$ and less than 1 s when $N = 5$ and 6. It also took less than about 1 min when $N = 3$. When $N = 2$ it took between 15 min and 65 h in the range measured, although this is not shown in the table because only part of it could be measured because it was too time-consuming. The amount of memory required is 82MiB maximum when $N = 3$ and is 32MiB when $N \geq 4$.

As the search time increases rapidly with $N < 4$, the number of ciphertext pairs required in this paper is defined to be N $(= 2^{n-2}) = 4$, though the key recovery is also feasible when $N = 3$ or $N = 2$. When $n = 4$, the required number of chosen plaintexts is $2^{n+4} = 2^8$. We counted the number of the function FO calculations and measured the amount of memory used during the search. The number of the function FO calculations was $\lesssim 2^{32}$, then the time complexity is $\lesssim 2^{29}$ in terms of one encryption which contains $8 = 2^3$ function FOs, and the memory used was about 2^{25} bytes (= 32MiB).

5 Summary

In this paper, we first described how to derive the weak keys by analyzing the key scheduling part and showed that when a weak key and the related key are used, there is a differential characteristic with a very large differential probability in the data randomizing part of the MISTY1 in the related-key setting.

Next, we described the attack procedures of distinguishing MISTY1 from a random permutation and the key recovery using this differential characteristic, and show that the computational complexity for these attacks is realistically feasible. For distinguishing MISTY1 from a random permutation, data and time complexity is 2^5 chosen plaintexts and the time for comparing 2^9 ciphertext pairs. For the key recovery, data, memory and time complexity is 2^8 chosen

plaintexts, 2^{25} bytes and the time for encrypting $\lesssim 2^{29}$ times. The actual time measured on a desktop PC was less than or equal about 2 s on average, although it depends on the value of secret keys and ciphertexts.

It should be noted that the set of the weak keys occupies a small fraction of the entire key space and that this attack is the related-key attack, so it is not considered to be a realistic threat in normal encryption applications.

References

1. Matsui, M.: New block encryption algorithm MISTY. In: Biham, E. (ed.) FSE 1997. LNCS, vol. 1267, pp. 54–68. Springer, Heidelberg (1997). https://doi.org/10.1007/BFb0052334
2. CRYPTREC Homepage. http://www.cryptrec.go.jp/en/method.html
3. Specifications in 2001 MISTY1. http://www.cryptrec.go.jp/en/cryptrec_03_spec_cypherlist_files/PDF/05_02espec.pdf
4. NESSIE Homepage. https://www.cosic.esat.kuleuven.be/nessie/
5. ISO/IEC Homepage. JTC1: ISO/IEC 18033: Security techniques - encryption algorithms - part 3: Block ciphers (2005)
6. Ohta, H., Matsui, M.: A description of the MISTY1 encryption algorithm (2000). https://tools.ietf.org/html/rfc2994
7. Lu, J., Yap, W.-S., Wei, Y.: Weak keys of the full MISTY1 block cipher for related-key differential cryptanalysis. In: Dawson, E. (ed.) CT-RSA 2013. LNCS, vol. 7779, pp. 389–404. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36095-4_25
8. Lu, J., Yap, W.S., Wei, Y.: Weak keys of the full MISTY1 block cipher for related-key amplified boomerang cryptanalysis. IET J. **12**(5), 389–397 (2018)
9. Todo, Y.: Integral cryptanalysis on full MISTY1. In: Gennaro, R., Robshaw, M. (eds.) CRYPTO 2015. LNCS, vol. 9215, pp. 413–432. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-47989-6_20
10. Todo, Y.: Integral cryptanalysis on full MISTY1. J. Cryptology. **30**, 920–959 (2017)
11. Bar-On, A., Keller, N.: A 2^{70} attack on the full MISTY1. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016. LNCS, vol. 9814, pp. 435–456. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53018-4_16
12. NIST FIPS PUB 46-3.: Data Encryption Standard (DES). http://csrc.nist.gov/csrs/media/publications/fips/46/3/archive/1990-10-25/documents/fips46-3.pdf
13. NIST FIPS PUB 197.: Advanced Encryption Standard (2001). http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf
14. Biham, E.: New types of cryptanalytic attacks using related keys. J. Cryptology **7**(4), 229–246 (1994)
15. Knudsen, L.R.: Cryptanalysis of LOKI 91. In: Seberry, J., Zheng, Y. (eds.) AUSCRYPT 1992. LNCS, vol. 718, pp. 196–208. Springer, Heidelberg (1993). https://doi.org/10.1007/3-540-57220-1_62
16. Biryukov, A., Khovratovich, D.: Related-key cryptanalysis of the full AES-192 and AES-256. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 1–18. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-10366-7_1
17. Biryukov, A., Khovratovich, D., Nikolić, I.: Distinguisher and related-key attack on the full AES-256. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 231–249. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03356-8_14