Cryptanalysis of a Reduced Version of the Block Cipher $E2$

Mitsuru Matsui and Toshio Tokita

Information Technology R&D Center
Mitsubishi Electric Corporation
5-1-1, Ofuna, Kamakura, Kanagawa, 247, Japan
matsui@iss.isl.melco.co.jp, tokita@iss.isl.melco.co.jp

Abstract. This paper deals with truncated differential cryptanalysis of the 128-bit block cipher $E2$, which is an AES candidate designed and submitted by NTT. Our analysis is based on byte characteristics, where a difference of two bytes is simply encoded into one bit information "0" (the same) or "1" (not the same). Since $E2$ is a strongly byte-oriented algorithm, this bytewise treatment of characteristics greatly simplifies a description of its probabilistic behavior and noticeably enables us an analysis independent of the structure of its (unique) lookup table. As a result, we show a non-trivial seven round byte characteristic, which leads to a possible attack of $E2$ reduced to eight rounds without IT and FT by a chosen plaintext scenario. We also show that by a minor modification of the byte order of output of the round function — which does not reduce the complexity of the algorithm nor violates its design criteria at all —, a non-trivial nine round byte characteristic can be established, which results in a possible attack of the modified $E2$ reduced to ten rounds without IT and FT, and reduced to nine rounds with IT and FT. Our analysis does not have a serious impact on the full $E2$, since it has twelve rounds with IT and FT; however, our results show that the security level of the modified version against differential cryptanalysis is lower than the designers' estimation.

1 Introduction

$E2$ [1] is a 128-bit block cipher designed by NTT, which is one of the fteen candidates in the rst round of the AES project. Its design criteria are conservative, adopting a Feistel network and a lookup table without shift and arithmetic operations (except multiplications in the initial transformation IT and the nal transformation FT). Moreover $E2$ has a strongly byte-oriented structure; all operations used in the data randomization phase are byte table lookups and byte xors except 32-bit multiplications in IT and FT, which successfully makes $E2$ a fast software cipher independent of target platforms, e.g. 8-bit microprocessors to 64-bit RISC computers.

This byte-oriented structure motivates us to cryptanalyze $E2$ by initially looking at relationship between the number/location of input byte changes and that of output byte changes. For instance, one byte change of input of the round

L. Knudsen (Ed.): FSE'99, LNCS 1636, pp. 71–80, 1999.

function always results in ve or six byte change of its output, which is a part of the design criteria of $E2$. However, when we change plural input bytes, say two bytes simultaneously, it is possible that only three output bytes are changed with the remaining ve bytes unchanged.

Since the round function of $E2$ has eight input/output bytes, its bytewise change pattern can be represented by eight-bit information where a di erence of two bytes is encoded into one bit information "0" (the same) or "1" (not the same). In this paper we call this change pattern "byte characteristic" or simply characteristic. Due to this simpli cation, it is not hard to create a complete 256 256 characteristic distribution table that exhausts all possibilities of input/output byte change patterns of the round function.

The next step of our analysis is to establish byte characteristics of the whole cipher and to nd e ective ones. Since the characteristics consist of sixteen bits, even a complete search for the best characteristic is possible in terms of computational complexity. We have reached an "iterative" byte characteristic of $E2$, which is non-trivial up to seven rounds. This leads to a possible attack of $E2$ reduced to eight rounds without IT and FT using 2^{100} chosen plaintext message blocks.

Also, we show that by a minor modi cation of the byte order of the output of the round function — which corresponds to change of BRL function [1] but does not reduce the complexity of the algorithm nor violates its design criteria at all —, a non-trivial nine round byte characteristic can be obtained, which results in a possible attack of the modi ed $E2$ reduced to ten rounds without IT and FT, and reduced to nine rounds with IT and FT using 2^{94} and 2^{91} chosen plaintext message blocks, respectively.

It should be pointed out that our analysis does not make use of the structure information of the lookup table; that is, our results hold for any (bijective) lookup table. However we will e ectively use the fact that $E2$ has only one lookup table. This means that if $E2$ would have many di erent tables in its round function, our attack could be (sometimes) harder. We will state the reason of this phenomenon in a later section.

Our analysis does not have a serious impact on the full $E2$, since it has twelve rounds with IT and FT; however our results show that the security level of the modi ed version against di erential cryptanalysis is lower than the designers' general estimation, which is applicable to both of the real and the modi ed $E2$.

2 Preliminaries

Figure 1 shows the entire structure of $E2$. Figures 2 to 4 shows its round function, initial transformation and nal transformation, respectively. In these gures the broken lines show subkey information, where we do not treat its key scheduling part. The notations in these gures will be used throughout this paper. For the exact detail, see [1]. For readers' convenience, we give algebraic description of the variable d_i in the round function in terms of the intermediate values c_i as follows:

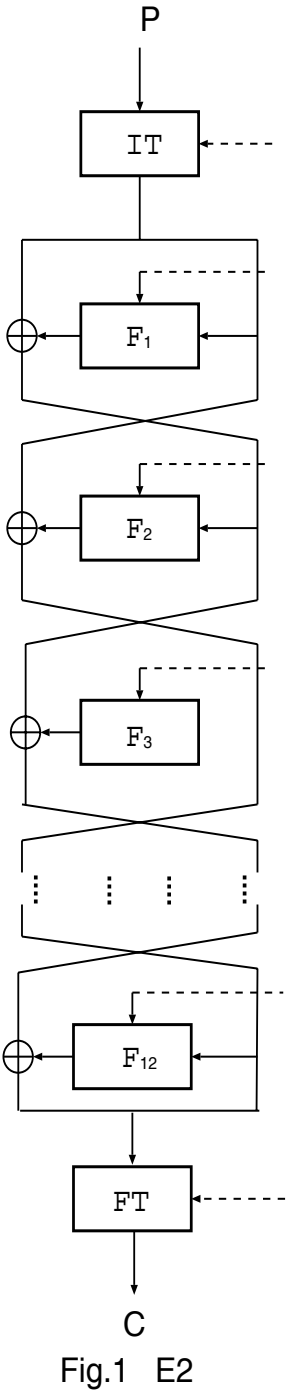

Fig.1 E2

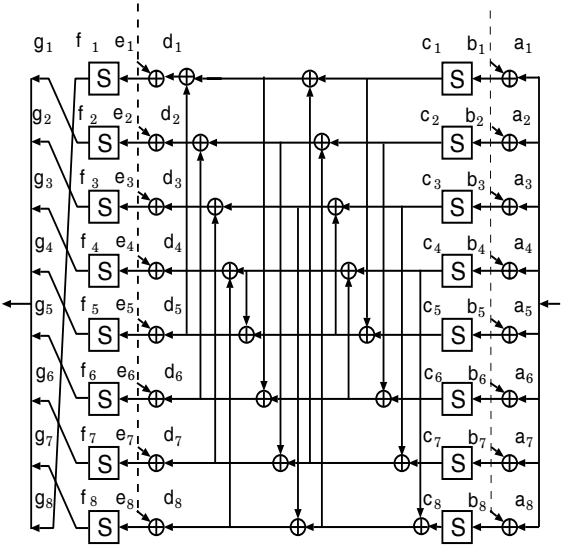

Fig.2 Round Function (F)

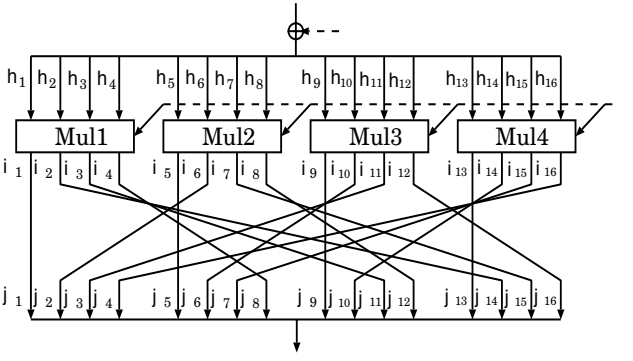

Fig.3 Initial Transformation (IT)

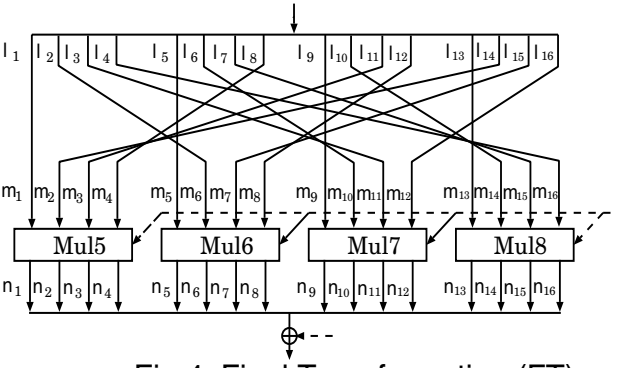

Fig.4 Final Transformation (FT)

$$
\begin{aligned}
d_1 &= c_2 \quad c_3 \quad c_4 \quad c_5 \quad c_6 \quad c_7 \\
d_2 &= c_1 \quad c_3 \quad c_4 \quad c_6 \quad c_7 \quad c_8 \\
d_3 &= c_1 \quad c_2 \quad c_4 \quad c_5 \quad c_7 \quad c_8 \\
d_4 &= c_1 \quad c_2 \quad c_3 \quad c_5 \quad c_6 \quad c_8 \\
d_5 &= c_1 \quad c_2 \quad c_4 \quad c_5 \quad c_6 \\
d_6 &= c_1 \quad c_2 \quad c_3 \quad c_6 \quad c_7 \\
d_7 &= c_2 \quad c_3 \quad c_4 \quad c_7 \quad c_8 \\
d_8 &= c_1 \quad c_3 \quad c_4 \quad c_5 \quad c_8
\end{aligned}
$$

3 Byte Characteristic Distribution of the Round Function

$E2$ was designed so that for any one byte change of input of the round function, at least ve output bytes (speci cally ve or six bytes) can be changed. For instance, it is easy to check that if we change a_1, leaving the remaining seven bytes unchanged, then g_1, g_2, g_3, g_4, g_5 and g_7 are always changed while the remaining two bytes are never changed.

Clearly this pattern of byte location does not depend on the amount of change of a_1. We describe this transition rule as follows:

$$(10000000) \to (11111010) \qquad p = 1. \tag{1}$$

Next, when we change two input bytes of the round function simultaneously, it is also easy to see that there are exactly two cases of output byte di erence patterns. For example, when we change a_1 and a_5 simultaneously, if the amount of change of c_1 (c_1) is equal to that of c_5 (c_5), then only three bytes g_1, g_5 and g_8 are changed, otherwise all bytes except g_6 are changed. Assuming that the input value (a_1 to a_8) and the amount of change (a_1 and a_5) are given randomly, the rst case happens with approximate probability $2^{\otimes 8}$ (the exact value is $1/255$, but for simplicity we use this approximation throughout this paper). The following denotes this transition rule:

$$(10001000) \to (10001001) \qquad p = 2^{\otimes 8}, \tag{2}$$

$$(10001000) \to (11111011) \qquad p = 1 \otimes 2^{\otimes 8}. \tag{3}$$

Similarly we can apply this notation to an arbitrary number of byte changes. Now one of the most useful byte characteristics of the round function of $E2$ is the following "cyclic" one, whose input pattern is the same as output pattern. This characteristic takes place when $c_1 =$ $c_4 =$ c_6, hence with the probability $(2^{\otimes 8})^2 = 2^{\otimes 16}$:

$$(10010100) \to (10010100) \qquad p = 2^{\otimes 16}. \tag{4}$$

Also, the following characteristic will be used in a later section:

$$(10110000) \to (10000010) \qquad p = 2^{\otimes 16}. \tag{5}$$

4 Byte Characteristic Distribution of $E2$

Using the cyclic characteristic shown in (4), we can obtain a seven round characteristic of $E2$ (without IT and FT) as shown in gure 5. Note that an xor operation outside the round function may cancel di erences; that is $1 \quad 1 = 0$ with probability 1/255 and $1 \quad 1 = 1$ with probability 254/255. For simplicity again, we will regard these probabilities as $2^{\otimes 8}$ and 1, respectively. In gure 5, this cancellation happens three times (three bytes) at an xor operation after the sixth round function. As a result, the seven round characteristic holds with approximate probability $(2^{\otimes 16})^5 \quad 2^{\otimes 24} = 2^{\otimes 104}$.

This means that when we change the rst, fourth and sixth bytes of the plaintext block simultaneously without changing other bytes, then after the seventh round, the probability that the three bytes of the same location change and other bytes do not change becomes $2^{\otimes 104}$. On the other hand, if the round function is a random function, the same change is expected to appear with probability $(2^{\otimes 8})^{13} = 2^{\otimes 104}$ again, since the number of unchanged bytes is thirteen. Therefore the expected number of "correct pairs" is the same as that of "wrong pairs".

Now remember that the correct pairs can be detected with probability $2^{\otimes 104}$ under the assumption that the amount of di erences of the speci ed input bytes (a_1, a_4 and a_6 in this case) are given randomly. However if we are able to give plaintext pairs with non random di erences in a chosen plaintext scenario, this probability may be greater. In fact when we generate input plaintext pairs such that the equation $a_1 = a_4 = a_6$ holds, then the transition probability of the second round function jumps to approximately $9.3 \quad 2^{\otimes 16}$, not $2^{\otimes 16}$, which is an experimental result. The reason of this increase is based on the fact that the following probability is signi cantly larger than $2^{\otimes 8}$ when $x = y$, while it is expected to be $2^{\otimes 8}$ in average when $x \neq y$.

$$P(\quad x, \quad y) \stackrel{def}{=} Prob_{x,y}\{S(x) \quad S(x \qquad x) = S(y) \quad S(y \qquad y)\}. \tag{6}$$

The exact probability of $P(\quad x, \quad y)$ depends on the structure of the substitution table S, but it is easy to prove that for any S, $P(\quad x, \quad y)$ is larger than $2^{\otimes 8}$ when $x = y$. Also it should be pointed out that this phenomenon can be utilized in our analysis of $E2$ because $E2$ has only one substitution table in its round function. If the function S of the left hand side di ers from that of the right hand side in the above de nition, the distribution of $P(\quad x, \quad y)$ will be "at", independent of x and y.

5 Possible Scenarios of an Attack of $E2$

5.1 $E2$ Reduced to Seven Rounds without IT and FT

The discussions in the previous section show that for $E2$ reduced to seven rounds, when 2^{104} chosen plaintext pairs such that $P_1 = P_4 = P_6$ are given, the expected number of ciphertext pairs that have the di erence pattern (10010100

00000000) is 9.3, where P_1, P_4 and P_6 denote the rst, the fourth and the sixth byte of plaintext, respectively. Note that these plaintext pairs can be obtained using 2^{97} plaintexts message blocks (97=104-8+1); for instance, they are given as the direct product set of all possible 2^{64} patterns of the left half of plaintexts and arbitrarily chosen 2^{33} patterns of the right half.

On the other hand, for a random permutation with the same chosen plaintext pairs, the expected number of ciphertext pairs that have the di erence pattern (10010100 00000000) is 1. This leads to the following scenario for distinguishing $E2$ reduced to seven rounds from a random permutation:

For a given cipher, if the number of ciphertext pairs that have the di erence pattern (10010100 00000000) is equal to or greater than a pre-determined value t, regard it as E2 reduced to seven rounds, otherwise regard it as a random permutation.

To estimate an appropriate value for t, we need the following lemma, which can be easily proven:

Lemma 1. *When a trial where an event occurs with probability p is carried out n/p times, assuming p is su ciently close to* 0 *and i is su ciently smaller than n/p, the probability that the event occurs exactly i times is*

$$(e^{\otimes n} n^i)/i!. \tag{7}$$

Using this lemma, we see that t=4 can be adopted in our case; for $E2$ reduced to seven rounds, the probability that the number of ciphertext pairs having the di erence pattern (10010100 00000000) is equal to or greater than four is 98%, while for a random permutation, the probability is expected to be 2%.

5.2 *E*2 Reduced to Eight Rounds without IT and FT

By applying again the seven round characteristic to the rst seven rounds of $E2$ reduced to eight rounds without IT and FT, we can narrow down the possibilities of subkey of the nal (the eighth) round using the following algorithm:

For each candidate for subkey of the nal round, decrypt all ciphertext pairs by one round. Then if the number of pairs that have the di erence pattern (10010100 00000000) after the seventh round is less than a pre-determined value t, discard the candidate as a wrong subkey.

Now let us use $2^{107} = 8 \quad 2^{104}$ chosen plaintext pairs such that $P_1 = P_4 = P_6$. Then if the candidate is the correct subkey, the expected number of pairs that have the di erence pattern (10010100 00000000) after the seventh round is $8 \quad 9.3 = 74.4$; however if it is a wrong subkey, the number of pairs is expected to be 8. Note that these plaintext pairs can be obtained using 2^{100} plaintexts message blocks (100=107-8+1).

A direct calculation using lemma 1 shows that $t = 60$, for instance, can sufficiently narrow down the possibilities of subkey of the final round. Specifically, for the correct subkey, the probability that the number of pairs having the difference pattern (10010100 00000000) after the seventh round is equal to or greater than 60 is 96%, while for the wrong subkey, the probability is expected to be $2^{\otimes 103}$.

The straightforward method for realizing the algorithm above requires complexity more than 2^{128}, but by discarding impossible pairs and introducing a counting method for the second layer subkey with 2^{64} counters, we can reduce the complexity to less than 2^{128}.

5.3 *E2* with a Modified Round Function

Our computer program has found that the seven round characteristic shown in figure 5 is the best one in the sense that it attains the maximal number of rounds with non-trivial probability. In this subsection, we try to find better characteristics by modifying the round function without violating its design criteria. Figure 7 is the modified round function we propose here. This modification — reordering output bytes of the round function, which is called BRL function — does not eliminate any original operations nor violates design criteria of $E2$.

This modified round function has the following "good" characteristics that correspond to equations (2) and (5) in the original round function, respectively:

$$(10001000) \rightarrow (10110000) \qquad p = 2^{\otimes 8}, \tag{8}$$

$$(10110000) \rightarrow (10001000) \qquad p = 2^{\otimes 16}. \tag{9}$$

Figure 6 shows a nine round characteristic which holds with probability $(2^{\otimes 8})^4 (2^{\otimes 16})^3 \; 2^{\otimes 24} = 2^{\otimes 104}$, while for a random round function the probability is expected to be $(2^{\otimes 8})^{14} = 2^{\otimes 112}$, which is significantly smaller. Therefore in a similar way to the previous subsection, we can extract subkey information of the final (the tenth) round of the modified $E2$ reduced to ten rounds without IT and FT.

The number of required plaintext pairs is 2^{109}, which can be generated from 2^{94} plaintext message blocks (94=109-16+1). Note that in this case we do not have to choose special plaintexts since the probability that correct pairs are detected is much larger than the probability that wrong pairs appear. An example of an appropriate value for t is 20; for the correct subkey of the final (the tenth) round, the probability that the number of pairs having the difference pattern (10001000 00000000) after the ninth round is equal to or greater than 20 is 99%, while for the wrong subkey, the probability is expected to be $2^{\otimes 121}$.

Lastly let us consider the modified $E2$ reduced to nine rounds **with** IT and FT. In IT and FT, 32-bit multiplications with subkey are used. However, since this multiplication is modulo 2^{32}, upper 32-bit of the resultant 64-bit information is simply discarded. Hence this multiplication has the following trivial byte characteristic:

$$(1000) \rightarrow (1000) \qquad p = 1. \tag{10}$$

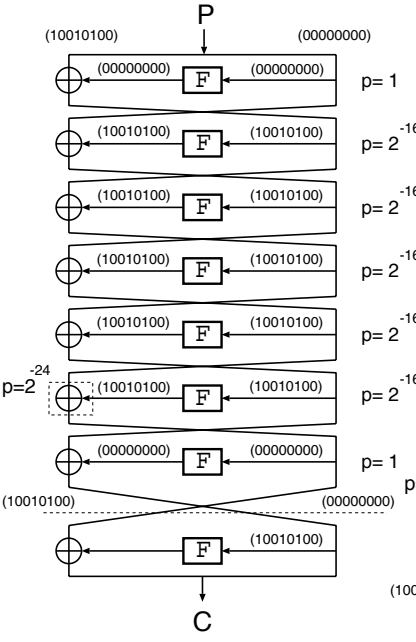

Fig.5 E2 reduced to eight rounds

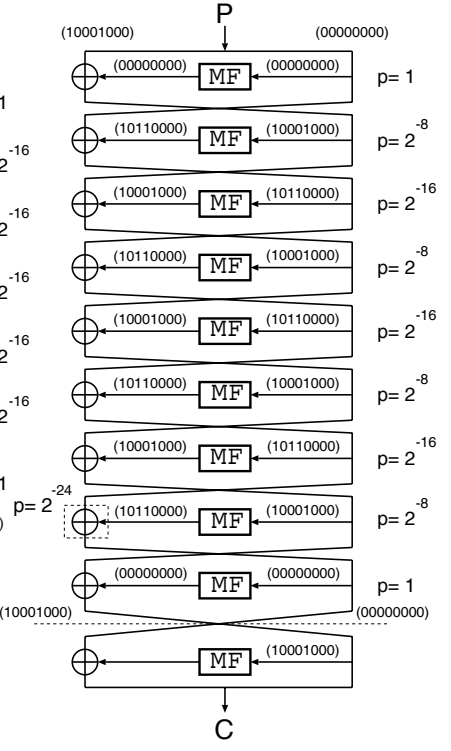

Fig.6 Modified E2 reduced to ten rounds

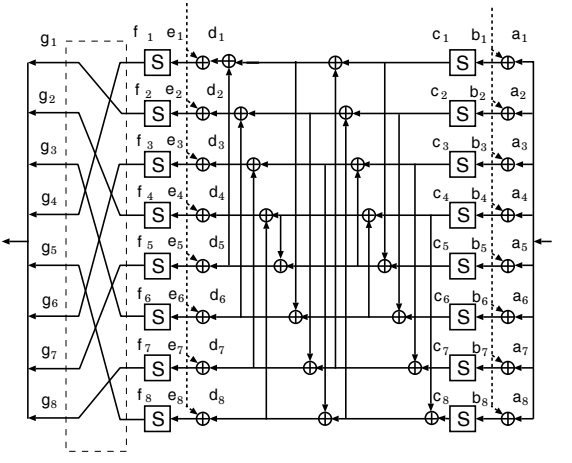

Fig.7 Modified Round Function (MF)

It follows from equation (10) that the characteristic shown in gure 6 can skip the IT and FT with probability 1. Therefore we have the following characteristic connecting a plaintext block and a ciphertext block direct:

$$(10001000\ \ 00000000) \rightarrow (10001000\ \ 00000000) \qquad p = 2^{\otimes 104}. \quad (11)$$

This means that in a chosen plaintext scenario, we can distinguish the modi ed $E2$ reduced to nine rounds with IT and FT from a random permutation. Speci cally, create 2^{106} plaintext pairs with the di erence pattern (10001000 00000000) from 2^{91} plaintext message blocks (91=106-16+1) and encrypt them. Then if a ciphertext pair that has the di erence pattern (10001000 00000000) is found, regard it as the modi ed $E2$ reduced to nine rounds with IT and FT, otherwise regard it as a random permutation (t=1). For $E2$ reduced to nine rounds with IT and FT, the probability that at least one ciphertext pair has the di erence pattern (10001000 00000000) is 98%, while for a random permutation, the probability is expected to be only 2%.

6 Discussions and Conclusions

It is easily seen that the e ectiveness of a byte characteristic can be evaluated by e = (hamming weight of the byte di erence pattern of the ciphertext pair) $\otimes$ log_2 (the characteristic probability). If m exceeds 16, the characteristic is not applicable to our analysis.

We wrote a computer program for searching the best byte characteristic of the modi ed $E2$ for all possible ($8! = 40320$) choices of the BRL function. The following is the summary of the search:

maximal effective number of rounds of the best characteristic	effectiveness e	number of choices of the BRL function
7	16	27688
7	15	8760
7	14	976
9	15	2896

Table 1: The best characteristic and the number of choices of the BRL function [1]

The designers of $E2$ have conjectured that the best nine round (ordinary bitwise) characteristic probability of $E2$ is much smaller than $2^{\otimes 140.34}$; their evaluation methodology does not depend on a choice of BRL function [1].

Our analysis shows that for most cases (maximal e ective number of rounds = 7), including the real $E2$, this estimation works well. However for the remaining 2896 cases, we can explicitly show a nine round bitwise di erential (not characteristic) whose probability is bigger than $2^{\otimes 120}$, which is signi cantly larger than the designers estimation. This indicates that in a byte-oriented algorithm, we should be careful of existence of detectable di erentials with high probability.

[1] After the publication of an earlier version of this paper, Shiho Moriai [5] showed a better attack of the real $E2$ based on another seven-round byte characteristic, whose effectiveness is 15; we confirmed that this is the real best byte characteristic of $E2$.

References

1. NTT-Nippon Telegraph and Telephone Corporation: E2 : Efficient Encryption algorithm. http://info.isl.ntt.co.jp/e2
2. Biham,E.,Shamir,A.: Differential Cryptanalysis of the Data Encryption Standard. Springer Verlag (1993)
3. Knudsen,L.R,Berson,T.A.: Truncated Differentials of SAFER. Third International Workshop of Fast Software Encryption, Lecture Notes in Computer Science 1039, Springer-Verlag(1996).
4. Lai,X.,Massey,J.L.,Murphy,S.: Markov Ciphers and Differential Cryptanalysis. Advances in Cryptology -Eurocrypt'91, Lecture Notes in Computer Science 547, Springer-Verlag(1991).
5. Moriai,S.: A talk at rump session in sixth international workshop of Fast Software Encryption (1999).