Differential Fault Analysis against AES-192 and AES-256 with Minimal Faults

Chong Hee KIM
Information Security Group, Université Catholique de Louvain,
Place Saint Barbe, 2, Louvain-la-Neuve, 1348, Belgium,
chong-hee.kim@uclouvain.be

***Abstract*—The naive implementation of AES is known to be vulnerable to *Differential Fault Analysis* (DFA). We can find the key of AES-128 (AES with 128-bit key) with one pair of correct and faulty ciphertexts. Recently several works on the extension of the attack to AES with 192 and 256-bit key have been published.**

Due to the longer key size and the characteristic of AES key schedule, we need subtle caution in attacking AES-192 and AES-256. We propose new DFA against AES with 192 and 256-bit key. We could retrieve AES-192 key with two pairs of correct and faulty ciphertexts. With three pairs we could succeed in finding the key of AES-256. These are the minimal faults among the existing methods.

***Keywords*-Fault attack; Differential Fault Analysis; AES; DFA;**

I. Introduction

Differential Fault Analysis (DFA) uses differential information between correct and faulty ciphertexts to figure out the secret key. Normally the attacker can get faulty ciphertexts by giving an external impact on the device with voltage variation, glitch, laser, etc [3].

The first DFA was presented by Biham and Shamir in 1997 [5]. Its target was DES [1]. Afterward many people tried to break several cryptosystems including AES [2]. Piret and Quisquater showed for the first time that few faults are enough to find the key of AES-128 (AES with 128-bit key) with a practical fault model [10]. Their method needs just two pairs of correct and faulty ciphertexts and assumes that a byte of the AES state is corrupted. In 2009, Fukunaga and Takahashi showed that the key of AES-128 could be deduced with one pair of correct and faulty ciphertexts and an exhaustive search of 2^{32} candidates [6]. The computational time was 8 - 35 minutes at Core2 Duo 3.0 GHz PC. Mukhopadhyay also showed the same result [9]. In 2010, Tunstall and Mukhopadhyay showed that an exhaustive search could be further reduced to 2^8 with one fault [12].

The work by Piret and Quisquater can be easily extended to attack AES with 192 and 256-bit key (AES-192 and AES-256 respectively). It needs four pairs of correct and faulty ciphertexts. Research on the DFA against AES-192 and AES-256 to reduce the required number of faulty pairs has recently started. The DFA against AES-128 tries to find the last subkey and then the master secret key can be directly computed from it through key schedule. However we cannot figure out the master secret key from the last subkey of AES-192 and AES-256. We need to find out more subkeys by DFA.

In 2009 Li et al. [7] proposed the first result on AES-192 and AES-256 based on the Moradi et al.'s DFA against AES-128 [8]. They found the secret key of AES-192 (and AES-256) with 16 or 3000 pairs of correct and faulty ciphertexts depending on the fault model. Barenghi et al. [4] showed that the secret key of AES-192 (and AES-256) could be retrieved with 16 pairs of correct and faulty ciphertexts. These two works require faulty ciphertexts with the *same* plaintext as well as quite many faults compared to DFA against AES-128.

They do not consider the feature of AES key schedule and therefore the relation between subkeys. Takahashi and Fukunaga showed that they could retrieve 192-bit key using three pairs of correct and faulty ciphertexts and 256-bit key using two pairs of correct and faulty ciphertexts and two pairs of correct and faulty plaintexts [11]. They could reduce the required number of faults by the detail analysis of AES key schedule and the use of an exhaustive search within practical computational time. However the additional requirement of faulty plaintexts (therefore access to a decryption oracle) in the attack against AES-256 seems undesirable.

In this paper, we propose new DFA against AES-192 and AES-256. We can retrieve AES-192 key with two pairs of correct and faulty ciphertexts and AES-256 key with three pairs of correct and faulty ciphertexts. Therefore our methods show the best performance among the existing works.

In the next section we briefly describe AES. Our fault model is presented in Section III. The basic concept of DFA against AES is given in Section IV. Section V and VI explain our proposed DFA against AES-192 and AES-256 respectively. After comparing with existing attacks in Section VII, we conclude in Section VIII.

II. AES

AES [2] can encrypt and decrypt 128 bits of block with 128, 192, or 256 bits of key. The intermediate computation result of AES, called *State*, is usually represented by a 4×4 matrix, each cell of which is a byte. For example, $S^i_{j,k}$ denotes $(j+1)^{th}$ row and $(k+1)^{th}$ column byte of i^{th} State, where $j, k \in \{0, \ldots 3\}$.

978-0-7695-4169-3/10 $26.00 © 2010 IEEE
DOI 10.1109/FDTC.2010.10

AES-128, AES-192, and AES-256 has 10, 12, and 14 rounds respectively. Each round function is composed of 4 transformations except the last round: *SubBytes*, *ShiftRows*, *MixColumns*, and *AddRoundKey*. The last round is lacking *MixColumns*. Our attack focuses on the last rounds.

1) SubBytes: It is made up of the application of 16 identical 8×8 S-boxes. This is a non-linear byte substitution. We denote the function of *SubBytes* $\mathbf{SB}$. That is, $\mathbf{SB}(S^i)$ = SubBytes(S^i). We denote *Inverse SubBytes* $\mathbf{SB^{-1}}$.

2) ShiftRows: Each row of the *State* is cyclically shifted over different offsets. Row 0 is not shifted, row 1 is shifted by 1 byte, row 2 is shifted by 2 bytes, and row 3 by 3 bytes. We denote *ShiftRows* and its inverse, *InverseShiftRows*, $\mathbf{SR}$ and $\mathbf{SR^{-1}}$ respectively.

3) MixColumns: This is a linear transformation to each column of the *State*. Each column is considered as polynomial over $\mathbb{F}_{2^8}$ and multiplied modulo $x^4 + 1$ with a fixed polynomial $a(x) = 03 * x^3 + 01 * x^2 + 01 * x + 02$. We denote the function of *MixColumns* $\mathbf{MC}$.

4) AddRoundKey: It is a bitwise XOR with a round key.

III. Fault model

We assume that a byte of the AES intermediate state is corrupted by fault injection. The corrupted value is random and unknown to the attacker.

The information on which is corrupted among 16 bytes may be known to the attacker. In [6], it was shown that a one-byte fault into the intermediate state on precise round and position can be easily injected by appropriate control of time slot of injection. If the attacker does not know the position of the corrupted byte, she can conduct 16 independent equivalent analysis. Therefore she needs 16 times more computation.

Finally we assume that the attacker can get a pair of correct and faulty ciphertexts.

IV. Basic concept

We introduce a concept of the differential fault analysis against AES-128 based on the Piret and Quisquater's method in [10], which is the basis of many variants published afterward. We assume that the attacker induces a fault between 7th and 8th *MixColumns* and gets a faulty ciphertext. The induced fault is assumed to corrupt one byte of the AES state as shown in Fig.1. We also assume that the attacker knows which byte is corrupted.

Then the attacker makes *differential equations* at the input of the 10th *SubBytes* with correct and faulty ciphertexts.

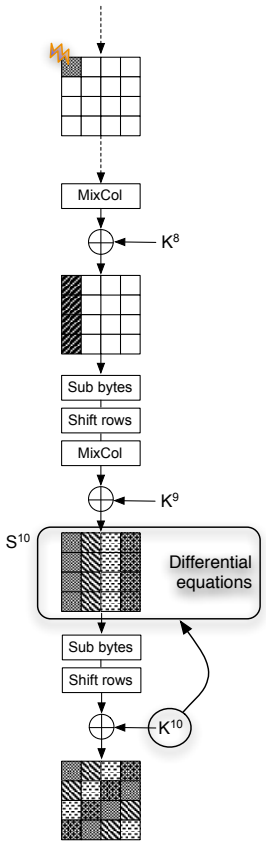


Figure 1. Basic concept of the differential fault analysis against AES-128

Each byte of the first column of S^{10} satisfies the following equations:

$$\begin{aligned} \Delta S^{10}_{(0,0)} &= 2\sigma, \\ \Delta S^{10}_{(1,0)} &= \sigma, \\ \Delta S^{10}_{(2,0)} &= \sigma, \\ \Delta S^{10}_{(3,0)} &= 3\sigma. \end{aligned}$$

Where σ is an unknown value $\in \{1, \ldots, 255\}$ and $\Delta S^{10} = S^{10} \oplus S^{*10}$. We can rewrite the above equations with 10th subkey and a pair of correct and faulty ciphertexts (C, C^*):

$$\mathbf{SB^{-1}}(C_{0,0} \oplus K^{10}_{0,0}) \oplus \mathbf{SB^{-1}}(C^*_{0,0} \oplus K^{10}_{0,0}) = 2\sigma, \quad (1)$$
$$\mathbf{SB^{-1}}(C_{1,3} \oplus K^{10}_{1,3}) \oplus \mathbf{SB^{-1}}(C^*_{1,3} \oplus K^{10}_{1,3}) = \sigma, \quad (2)$$
$$\mathbf{SB^{-1}}(C_{2,2} \oplus K^{10}_{2,2}) \oplus \mathbf{SB^{-1}}(C^*_{2,2} \oplus K^{10}_{2,2}) = \sigma, \quad (3)$$
$$\mathbf{SB^{-1}}(C_{3,1} \oplus K^{10}_{3,1}) \oplus \mathbf{SB^{-1}}(C^*_{3,1} \oplus K^{10}_{3,1}) = 3\sigma. \quad (4)$$

Among 2^{32} candidates for $< K^{10}_{0,0}, K^{10}_{1,3}, K^{10}_{2,2}, K^{10}_{3,1} >$, in average 2^8 candidates satisfy the above equations simultaneously. Because the probability of passing the above equations is $\frac{255}{255^4}$ and therefore the number of remaining candidates is $\frac{255}{255^4} \times 2^{32} \simeq 2^8$ [10].

We note that if we have t out of four equations then the probability of passing t equations is $\frac{255}{255^t}$. That is, if we have two equations the probability is about $\frac{1}{2^8}$.

For the other columns we can construct similar equations and therefore we have the sets of candidates for $< K^{10}_{0,0}, K^{10}_{1,3}, K^{10}_{2,2}, K^{10}_{3,1} >$, $< K^{10}_{0,1}, K^{10}_{1,0}, K^{10}_{2,3}, K^{10}_{3,2} >$, $< K^{10}_{0,2}, K^{10}_{1,1}, K^{10}_{2,0}, K^{10}_{3,3} >$, and $< K^{10}_{0,3}, K^{10}_{1,2}, K^{10}_{2,1}, K^{10}_{3,0} >$. As each set has 2^8 candidates we have 2^{32} candidates for the 10th subkey.

Then we can find the 10th subkey by computing the equations once more with another pair of correct and faulty ciphertexts and therefore reducing the number of candidates to one [10].

Otherwise we can find it with an exhaustive search of 2^{32} candidates [6], [9]. If the attacker does not know the location of the corrupted byte, then she can conduct 16 independent equivalent analysis. Therefore she has to perform $16 \times 2^{32} = 2^{36}$ computations. Later it was shown that an exhaustive search of 2^8 was enough [12].

With a current normal PC, an exhaustive search of 2^{32} can be done within tens of minutes [6]. Therefore we can use up to 2^{32} exhaustive search to minimize the required number of faults.

V. Differential fault analysis against AES-192

We propose two differential fault analysis against AES-192. Both require two pairs of correct and faulty ciphertexts. The former method needs a 2^{32} exhaustive search and the latter 2^8. The first method assumes that both faults are induced between 9th and 10th *MixColumns*. The second method needs one fault between 9th and 10th *MixColumns* and the other between 8th and 9th *MixColumns*.

A. Attack procedure 1

1) Obtain two pairs of correct and faulty ciphertexts (C_1, C_1^*) and (C_2, C_2^*). Where the faults are injected between 9th and 10th *MixColumns*
2) Find K^{12}.
3) Find the left-half of K^{11} with key schedule.
4) Find 2^{32} candidates for the right-half of K^{11}.
5) Find the master secret key with an exhaustive search of 2^{32}.

1) Find K^{12}: With two pairs of correct and faulty ciphertexts, we can find the last subkey K^{12} using the method described in Section IV.

As in Fig. 2(a), we first construct differential equations at the input of the 12th *SubBytes*. Then we exclude candidates for K^{12} that do not satisfy the differential equations. We can finally find K^{12} using two pairs of correct and faulty ciphertexts.

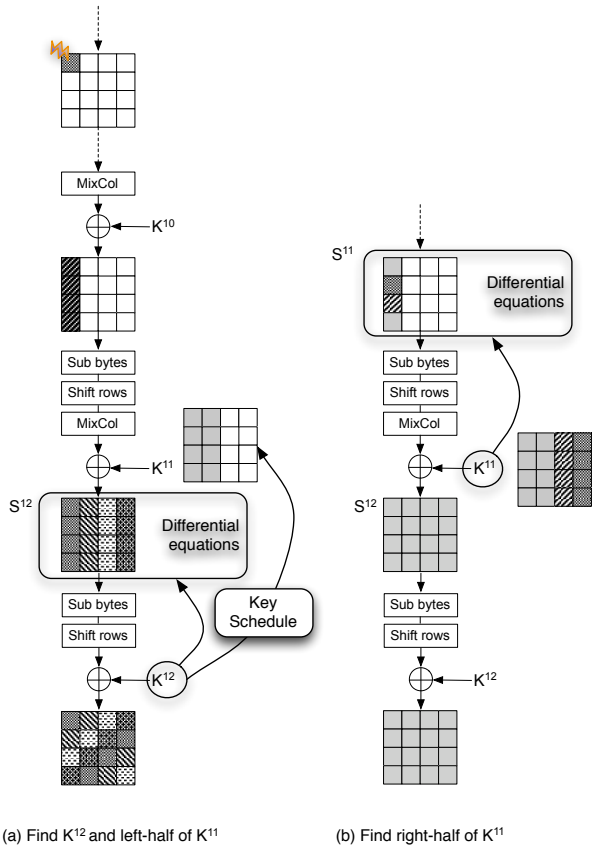


Figure 2. Differential fault analysis against AES-192: method 1

2) Find the left-half of K^{11} with key schedule: From the AES key schedule, we have the following equations:

$$\begin{aligned} K^{12}[4] &= K^{12}[3] \oplus K^{11}[2], \\ K^{12}[3] &= K^{12}[2] \oplus K^{11}[1]. \end{aligned}$$

Where $K^i[j]$ is the jth column of K^i, $j \in \{1, 2, 3, 4\}$.

As we know K^{12} from the previous step, we can find the left-half of K^{11}.

3) Find 2^{32} candidates for the right-half of K^{11}: We construct differential equations at the input of the 11th *SubBytes* as shown in Fig. 2(b):

$$\begin{aligned} 2\sigma = \; & \mathbf{SB^{-1}}(14(S^{12}_{0,0} \oplus K^{11}_{0,0}) \oplus 13(S^{12}_{1,0} \oplus K^{11}_{1,0}) \oplus \\ & 11(S^{12}_{2,0} \oplus K^{11}_{2,0}) \oplus 13(S^{12}_{3,0} \oplus K^{11}_{3,0})) \oplus \\ & \mathbf{SB^{-1}}(14(S^{*12}_{0,0} \oplus K^{11}_{0,0}) \oplus 13(S^{*12}_{1,0} \oplus K^{11}_{1,0}) \oplus \\ & 11(S^{*12}_{2,0} \oplus K^{11}_{2,0}) \oplus 13(S^{*12}_{3,0} \oplus K^{11}_{3,0})), \end{aligned} \quad (5)$$

$$\sigma = \mathbf{SB^{-1}}(9(S^{12}_{0,3} \oplus K^{11}_{0,3}) \oplus 14(S^{12}_{1,3} \oplus K^{11}_{1,3}) \oplus 11(S^{12}_{2,3} \oplus K^{11}_{2,3}) \oplus 13(S^{12}_{3,3} \oplus K^{11}_{3,3})) \oplus \mathbf{SB^{-1}}(9(S^{*12}_{0,3} \oplus K^{11}_{0,3}) \oplus 14(S^{*12}_{1,3} \oplus K^{11}_{1,3}) \oplus 11(S^{*12}_{2,3} \oplus K^{11}_{2,3}) \oplus 13(S^{*12}_{3,3} \oplus K^{11}_{3,3})), \quad (6)$$

$$\sigma = \mathbf{SB^{-1}}(13(S^{12}_{0,2} \oplus K^{11}_{0,2}) \oplus 9(S^{12}_{1,2} \oplus K^{11}_{1,2}) \oplus 14(S^{12}_{2,2} \oplus K^{11}_{2,2}) \oplus 11(S^{12}_{3,2} \oplus K^{11}_{3,2})) \oplus \mathbf{SB^{-1}}(13(S^{*12}_{0,2} \oplus K^{11}_{0,2}) \oplus 9(S^{*12}_{1,2} \oplus K^{11}_{1,2}) \oplus 14(S^{*12}_{2,2} \oplus K^{11}_{2,2}) \oplus 11(S^{*12}_{3,2} \oplus K^{11}_{3,2})). \quad (7)$$

Where $S^{12}_{i,j}$ and $S^{*12}_{i,j}$ can be computed from correct and faulty ciphertexts and K^{12}.

As we know the first column of K^{11}, $(K^{11}_{0,0}, K^{11}_{1,0}, K^{11}_{2,0}, K^{11}_{3,0})$, we can compute σ from Eq.(5). Then only the fourth column of K^{11}, $(K^{11}_{0,3}, K^{11}_{1,3}, K^{11}_{2,3}, K^{11}_{3,3})$, is unknown in Eq.(6). The probability that a candidate for $(K^{11}_{0,3}, K^{11}_{1,3}, K^{11}_{2,3}, K^{11}_{3,3})$ passes Eq.(6) is $\frac{1}{255}$. Therefore among 2^{32} candidates for $(K^{11}_{0,3}, K^{11}_{1,3}, K^{11}_{2,3}, K^{11}_{3,3})$, about $2^{32} \times \frac{1}{2^8}$ candidates satisfy this equation. As we have two pairs of correct and faulty ciphertexts we can reduce the candidates once more. Finally we have $2^{32} \times \frac{1}{2^{16}} = 2^{16}$ candidates.

The 2^{16} candidates for the third column of K^{11}, $(K^{11}_{0,2}, K^{11}_{1,2}, K^{11}_{2,2}, K^{11}_{3,2})$, can be computed similarly from Eq.(7). Therefore the number of candidates for the right-half of K^{11} is 2^{32}.

4) Find the master secret key with the exhaustive search of 2^{32}: We can compute the candidates for the master secret key with K^{12} and the candidates for K^{11}. Then we find the correct one by comparing the correct ciphertext and the ciphertexts calculated from the candidates for the master secret key.

B. Attack procedure 2

1) Induce a byte fault between 9th and 10th *MixColumns*. Obtain a pair of correct and faulty ciphertexts (C_1, C^*_1).
2) Induce a byte fault between 8th and 9th *MixColumns*. Obtain a pair of correct and faulty ciphertexts (C_2, C^*_2).
3) Find 2^{32} candidates for K^{12} with (C_1, C^*_1).
4) Compute the 2^{32} candidates for left-half of K^{11} with key schedule.
5) Reduce the number of candidates for K^{12} and the left-half of K^{11} to 2^{24}.
6) Find the left-half of K^{11} and K^{12} with (C_2, C^*_2).
7) Find the 2^8 candidates for right-half of K^{11} with (C_2, C^*_2).
8) Find the master secret key with an exhaustive search of 2^8.

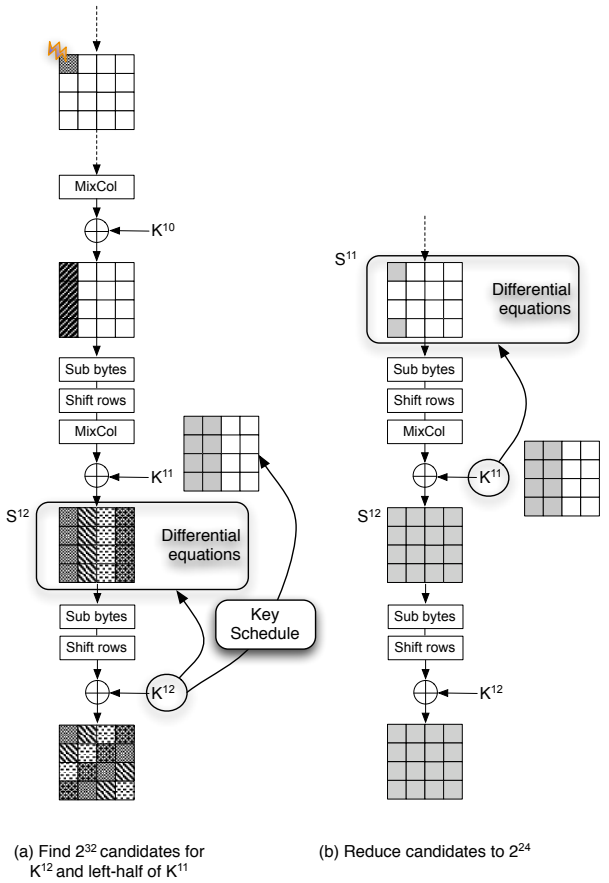


Figure 3. Differential fault analysis against AES-192: method 2, 1st fault

1) Find 2^{24} candidates for K^{12} and the left-half of K^{11} with (C_1, C^*_1): We construct the differential equations at the input of the 12th *SubBytes* as in Fig. 3(a). As described in Section IV, we can find 2^{32} candidates for K^{12} with (C_1, C^*_1). Furthermore we can find the 2^{32} candidates for the left-half of K^{11} with key schedule.

Then we construct the differential equations for the left-half of K^{11} at the input of the 11th *SubBytes* as in Fig. 3(b):

$$2\sigma = \mathbf{SB^{-1}}(14(S^{12}_{0,0} \oplus K^{11}_{0,0}) \oplus 13(S^{12}_{1,0} \oplus K^{11}_{1,0}) \oplus 11(S^{12}_{2,0} \oplus K^{11}_{2,0}) \oplus 13(S^{12}_{3,0} \oplus K^{11}_{3,0})) \oplus \mathbf{SB^{-1}}(14(S^{*12}_{0,0} \oplus K^{11}_{0,0}) \oplus 13(S^{*12}_{1,0} \oplus K^{11}_{1,0}) \oplus 11(S^{*12}_{2,0} \oplus K^{11}_{2,0}) \oplus 13(S^{*12}_{3,0} \oplus K^{11}_{3,0})), \quad (8)$$

$$\sigma = \mathbf{SB^{-1}}(11(S^{12}_{0,1} \oplus K^{11}_{0,1}) \oplus 13(S^{12}_{1,1} \oplus K^{11}_{1,1}) \oplus 9(S^{12}_{2,1} \oplus K^{11}_{2,1}) \oplus 14(S^{12}_{3,1} \oplus K^{11}_{3,1})) \oplus \mathbf{SB^{-1}}(11(S^{*12}_{0,1} \oplus K^{11}_{0,1}) \oplus 13(S^{*12}_{1,1} \oplus K^{11}_{1,1}) \oplus 9(S^{*12}_{2,1} \oplus K^{11}_{2,1}) \oplus 14(S^{*12}_{3,1} \oplus K^{11}_{3,1})). \quad (9)$$

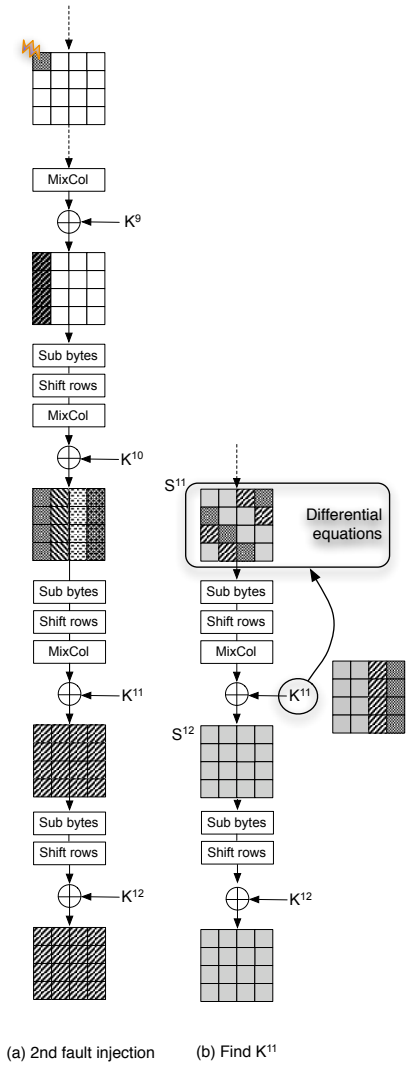


Figure 4. Differential fault analysis against AES-192: method 2, 2nd fault

The probability that one candidate for the left-half of K^{11} passes Eq.(8) and Eq.(9) is $\frac{255}{255^2}$. Therefore we have $2^{32} \times \frac{255}{255^2} \simeq 2^{24}$ candidates for the left-half of K^{11}. The number of candidates for K^{12} also reduces to 2^{24}.

2) Find the left-half of K^{11} and K^{12} with (C_2, C_2^*): From now on we use the second pair of correct and faulty ciphertexts. The fault induced between 8th and 9th *MixColumns* propagates as in Fig. 4(a). We construct the differential equations for the left-half of K^{11} at the input of the 11th *SubBytes* as in Fig. 4(b). At the first column of S^{11} we have:

$$\begin{cases} \Delta S^{11}_{(0,0)} = 2\sigma_1, \\ \Delta S^{11}_{(3,0)} = 3\sigma_1. \end{cases} \tag{10}$$

We note that only the first and the fourth bytes are related with the left-half of K^{11}. Similarly we can construct for the second, the third, and the fourth columns of S^{11}:

$$\begin{cases} \Delta S^{11}_{(0,1)} = \sigma_2, \\ \Delta S^{11}_{(1,1)} = \sigma_2, \end{cases} \tag{11}$$

$$\begin{cases} \Delta S^{11}_{(1,2)} = 3\sigma_3, \\ \Delta S^{11}_{(2,2)} = 2\sigma_3, \end{cases} \tag{12}$$

$$\begin{cases} \Delta S^{11}_{(2,3)} = \sigma_4, \\ \Delta S^{11}_{(3,3)} = \sigma_4. \end{cases} \tag{13}$$

All these equations include K^{12}, the left-half of K^{11}, and correct and faulty ciphertexts. Therefore we can construct equations similar to Eq.(5) - Eq. (9).

The probability that one candidate for K^{12} and the left-half of K^{11} passes Eq.(10) is $\frac{255}{255^2} \simeq \frac{1}{2^8}$. Therefore we have only one candidate that satisfy Eq.(10) - Eq.(13).

3) Find 2^8 candidates for right-half of K^{11} with (C_2, C_2^*): We construct differential equations for the first column of the input of the 11th *SubBytes* as shown in Fig. 4(b):

$$\begin{aligned} 2\sigma \;=\; & \mathbf{SB^{-1}}(14(S^{12}_{0,0} \oplus K^{11}_{0,0}) \oplus 13(S^{12}_{1,0} \oplus K^{11}_{1,0}) \oplus \\ & 11(S^{12}_{2,0} \oplus K^{11}_{2,0}) \oplus 13(S^{12}_{3,0} \oplus K^{11}_{3,0})) \oplus \\ & \mathbf{SB^{-1}}(14({S^*}^{12}_{0,0} \oplus K^{11}_{0,0}) \oplus 13({S^*}^{12}_{1,0} \oplus K^{11}_{1,0}) \oplus \\ & 11({S^*}^{12}_{2,0} \oplus K^{11}_{2,0}) \oplus 13({S^*}^{12}_{3,0} \oplus K^{11}_{3,0})), \end{aligned} \tag{14}$$

$$\begin{aligned} \sigma \;=\; & \mathbf{SB^{-1}}(9(S^{12}_{0,3} \oplus K^{11}_{0,3}) \oplus 14(S^{12}_{1,3} \oplus K^{11}_{1,3}) \oplus \\ & 11(S^{12}_{2,3} \oplus K^{11}_{2,3}) \oplus 13(S^{12}_{3,3} \oplus K^{11}_{3,3})) \oplus \\ & \mathbf{SB^{-1}}(9({S^*}^{12}_{0,3} \oplus K^{11}_{0,3}) \oplus 14({S^*}^{12}_{1,3} \oplus K^{11}_{1,3}) \oplus \\ & 11({S^*}^{12}_{2,3} \oplus K^{11}_{2,3}) \oplus 13({S^*}^{12}_{3,3} \oplus K^{11}_{3,3})), \end{aligned} \tag{15}$$

Where $S^{12}_{i,j}$ and ${S^*}^{12}_{i,j}$ can be computed from correct and faulty ciphertexts and K^{12}.

As we know the first column of K^{11}, we can compute σ from Eq.(14). Then only fourth column of K^{11}, $(K^{11}_{0,3}, K^{11}_{1,3}, K^{11}_{2,3}, K^{11}_{3,3})$, is unknown in Eq.(15). The probability that a candidate for $(K^{11}_{0,3}, K^{11}_{1,3}, K^{11}_{2,3}, K^{11}_{3,3})$ passes Eq.(15) is $\frac{1}{255}$. Therefore among 2^{32} candidates for $(K^{11}_{0,3}, K^{11}_{1,3}, K^{11}_{2,3}, K^{11}_{3,3})$, about $2^{32} \times \frac{1}{2^8}$ candidates satisfy this equation. As we can construct similar equations for the 2nd, 3rd, and 4th column of S^{11}, we can reduce the number of candidates.

We have at least 16 candidates for 4-byte key with one faulty ciphertext. Because $K^{11}[3]$ and $(K^{11}[3] \oplus S^{12}[3] \oplus {S^*}^{12}[3])$ and 14 other candidates when only some bytes of

$S^{12}[3] \oplus S^{*12}[3]$ are XORed to $K^{11}[3]$ are returned from the attack[1]. More detail analysis can be found in [10].

The candidates for $(K^{11}_{0,2}, K^{11}_{1,2}, K^{11}_{2,2}, K^{11}_{3,2})$ can be computed similarly. Therefore we have 2^8 candidates for right-half of K^{11}.

4) Find the master secret key with K^{11} and K^{12}: From K^{12} and 2^8 candidates for K^{11}, we can find the master secret key using the key schedule and an exhaustive search.

VI. Differential fault analysis against AES-256

A. Attack procedure

1) Obtain two pairs of correct and faulty ciphertexts (C_1, C_1^*) and (C_2, C_2^*) by giving faults between 11th and 12th *MixColumns*.
2) Obtain a pair of correct and faulty ciphertexts (C_3, C_3^*) by giving faults between 10th and 11th *MixColumns*.
3) Find K^{14} with (C_1, C_1^*) and (C_2, C_2^*).
4) Find 2^{32} candidates for K^{13} with (C_3, C_3^*).
5) Find the master secret key with an exhaustive search of 2^{32}.

1) Find K^{14} with (C_1, C_1^*) and (C_2, C_2^*): We construct differential equations at the input of 14th *SubBytes* (Fig. 5(a)) by using the method described in Section IV. With two pairs of correct and faulty ciphertexts we find K^{14} satisfying the differential equations.

2) Find 2^{32} candidates for K^{13} with (C_3, C_3^*): We construct the differential equations at the input of 13th *SubBytes* (Fig. 5(b)) based on the output of 13th *SubBytes*, M.

We note that the differential equations based on the K^{13} as in the previous sections can not be used anymore due to the 13th *MixColumns*.

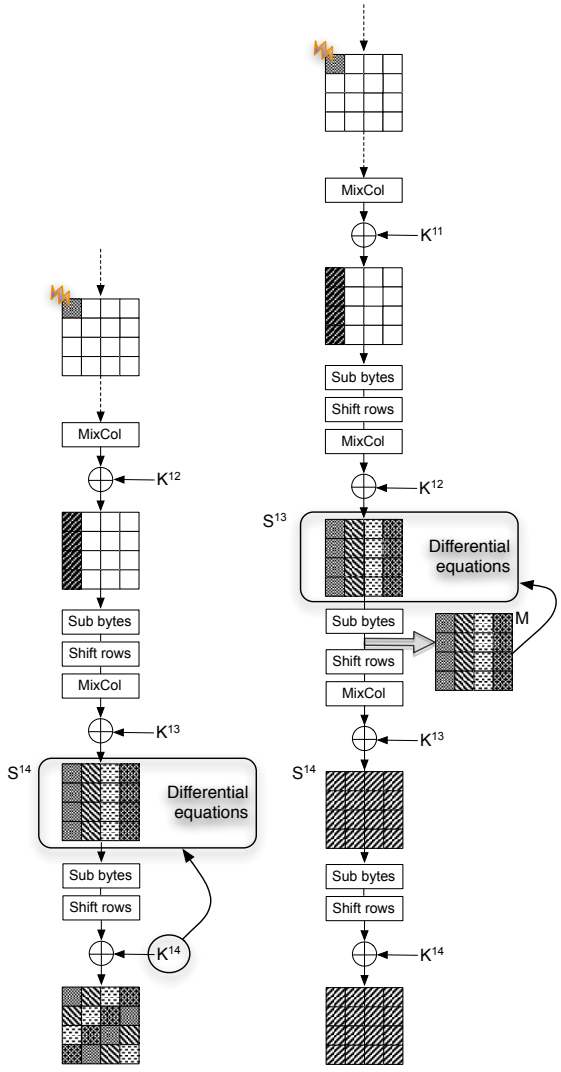
For the first column of S^{13} we have:

$$\mathbf{SB^{-1}}(M_{0,0}) \oplus \mathbf{SB^{-1}}(M_{0,0} \oplus \varepsilon_{0,0}) = 2\sigma, \quad (16)$$
$$\mathbf{SB^{-1}}(M_{0,1}) \oplus \mathbf{SB^{-1}}(M_{0,1} \oplus \varepsilon_{1,3}) = \sigma, \quad (17)$$
$$\mathbf{SB^{-1}}(M_{0,2}) \oplus \mathbf{SB^{-1}}(M_{0,2} \oplus \varepsilon_{2,2}) = \sigma, \quad (18)$$
$$\mathbf{SB^{-1}}(M_{0,3}) \oplus \mathbf{SB^{-1}}(M_{0,3} \oplus \varepsilon_{3,1}) = 3\sigma. \quad (19)$$

Where, $\varepsilon = InvMixColumns(S^{14} \oplus S^{*14})$. As we know K^{14}, we can compute the value of S^{14} and S^{*14} from correct and faulty ciphertexts respectively. The differential information between S^{14} and S^{*14} is safely transformed through the *InvMixColumns* since it is linear w.r.t. the XOR operation.

Among 2^{32} candidates for $< M_{0,0}, M_{0,1}, M_{0,2}, M_{0,3} >$, only $2^{32} \times \frac{255}{255^4} \simeq 2^8$ candidates satisfy the above equations.

[1]We note that in average only one candidate for left-half of K^{11} is left in the previous step since two pairs of correct and faulty ciphertexts are used for eliminating candidates (the first pair (C_1, C_1^*) is used to reduce candidates to 2^{24} and (C_2, C_2^*) for the final one).



(a) Find K^{14} with 2 faults (b) Find 2^{32} candidates of K^{13} with 1 fault

Figure 5. Differential fault analysis against AES-256

Similarly we can construct differential equations for the 2nd, 3rd, and 4th column. Therefore we have 2^{32} candidates for M.

Finally we can find the candidates for K^{13} from the following equation:

$$K^{13} = \mathbf{MC}(\mathbf{SR}(M)) \oplus S^{14}.$$

3) Find the master secret key with the exhaustive search of 2^{32}: With K^{14} and the 2^{32} candidates for K^{13} we can find the candidates for the master secret key. Then we can find the correct one by comparing the correct ciphertext and the ciphertexts calculated from the candidates for the master

Table I
COMPARISON WITH EXISTING ATTACKS ON AES-192

Reference	Fault model	No. of faults	Exhaustive search
[10]	1 byte random fault	4	1
[7] method 1	1-4 bytes random fault	$12^{\dagger}$	1
[7] method 2	4 bytes random fault	$3000^{\dagger}$	1
[4]	1 byte random fault	$16^{\dagger}$	1
[11]	1 byte random fault	3	2^8
Our attack 1	1 byte random fault	2	2^{32}
Our attack 2	1 byte random fault	2	2^8

†: with same plaintext

Table II
COMPARISON WITH EXISTING METHODS ON AES-256

Reference	Fault model	No. of faults	Exhaustive search
[10]	1 byte random fault	4	1
[7] method 1	1-4 bytes random fault	$12^{\dagger}$	1
[7] method 2	4 bytes random fault	$3000^{\dagger}$	1
[4]	1 byte random fault	$16^{\dagger}$	1
[11]	1 byte random fault	$4^{\ddagger}$	2^{13}
Our attack	1 byte random fault	3	2^{32}

†: with same plaintext
‡: 2 faulty plaintexts and 2 faulty ciphertexts

secret key.

VII. COMPARISON

The methods of Li et al. require 12 or 3000 faulty ciphertexts depending on the fault model to break AES-192 and AES-256 [7]. The same plaintext should be used for all faulty ciphertexts.

The method of Barenghi et al. requires 16 faulty ciphertexts with the same plaintext [4].

Takahashi and Fukunaga's method to break AES-192 requires 3 faulty ciphertexts and needs to search 2^8 candidates [11]. Their method to break AES-256 requires 2 pairs of correct and faulty ciphertexts and 2 pairs of correct and faulty plaintexts and searches 2^{13} candidates. And it needs to access to a decryption oracle to get faulty plaintexts.

The existing methods need either faulty ciphertexts with the same plaintext or a decryption oracle. Therefore it is less practical to use. Our methods do not have this restriction as well as need less faults. We compared our methods with existing ones in Table I and II.

VIII. CONCLUSIONS

We proposed new differential fault analysis against AES with 192 and 256-bit keys. We could retreive AES-192 key with two pairs of correct and faulty ciphertexts. With three pairs we could deduce the secret key of AES-256 within a practical time.

ACKNOWLEDGEMENT

This work is supported by the Belgian Walloon Region project TRASILUX.

REFERENCES

[1] National Institute of Standard and Technology, *Data Encryption Standard*, NIST FIPS PUB 46-2, 1993.

[2] National Institute of Standard and Technology, *Advanced Encryption Standard*, NIST FIPS PUB 197, 2001.

[3] H. Bar-El, H. Choukri, D. Naccache, M. Tunstall, and C. Whelan. The sorcerer's apprentice guide to fault attacks. In *Fault Diagnosis and Tolerance in Cryptography in association with DSN 2004 – The International Conference on Dependable Systems and Networks*, pages 330–342, 2004.

[4] A. Barenghi, G. Bertoni, L. Breveglieri, M. Pellicioli, and G. Pelosi. Low voltage fault attacks to AES and RSA on general purpose processors. IACR eprint archive, 2010-130.

[5] E. Biham and A. Shamir. Differential fault analysis of secret key cryptosystems. In *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference*, volume 1294 of *Lecture Notes in Computer Science*, pages 513–525. Springer, 1997.

[6] T. Fukunaga and J. Takahashi. Practical fault attack on a cryptographic LSI with IOS/IEC 18033-3 block ciphers. In *6th International Workshop on Fault Diagnosis and Tolerance in Cryptography, FDTC 2009*. IEEE Computer Society, 2009.

[7] W. Li, D. Gu, Y. Wang, J. Li, and Z. Liu. An extension of differential fault analysis on AES. In *International Conference on Network and System Security*, pages 443–446, Los Alamitos, CA, USA, 2009. IEEE Computer Society.

[8] A. Moradi, M. T. M. Shalmani, and M. Salmasizadeh. A generalized method of differential fault attack against AES cryptosystem. In *Cryptographic Hardware and Embedded Systems - CHES 2006, 8th International Workshop*, volume 4249 of *Lecture Notes in Computer Science*, pages 91–100. Springer, 2006.

[9] D. Mukhopadhyay. An improved fault based attack of the advanced encryption standard. In *AFRICACRYPT 2009*, volume 5580 of *Lecture Notes in Computer Science*, pages 421–434. Springer-Verlag, 2009.

[10] G. Piret and J.-J. Quisquater. A differential fault attack technique against SPN structures, with application to the AES and KHAZAD. In *CHES*, volume 2779 of *Lecture Notes in Computer Science*, pages 77–88. Springer, 2003.

[11] J. Takahashi and T. Fukunaga. Differential fault analysis on AES with 192 and 256-bit keys. IACR eprint archive, 2010-023.

[12] M. Tunstall and D. Mukhopadhyay. Differential fault analysis of the advanced encryption standard using a single fault. IACR eprint archive, 2009-575.