Related-Key Rectangle Attacks on Reduced AES-192 and AES-256 *

Jongsung Kim[1], Seokhie Hong[1], and Bart Preneel[2]

[1] Center for Information Security Technologies (CIST),
Korea University, Anam Dong, Sungbuk Gu, Seoul, Korea
{joshep,hsh}@cist.korea.ac.kr
[2] Katholieke Universiteit Leuven, Dept. ESAT/SCD-COSIC,
Kasteelpark Arenberg 10, B-3001 Heverlee, Belgium
Bart.Preneel@esat.kuleuven.be

Abstract. This paper examines the security of AES-192 and AES-256 against a related-key rectangle attack. We find the following new attacks: 8-round reduced AES-192 with 2 related keys, 10-round reduced AES-192 with 64 or 256 related keys and 9-round reduced AES-256 with 4 related keys. Our attacks reduce the complexity of earlier attacks presented at FSE 2005 and Eurocrypt 2005: for reduced AES-192 with 8 rounds, we decrease the required number of related keys from 4 to 2 at the cost of a higher data and time complexity; we present the first shortcut attack on AES-192 reduced to 10 rounds; for reduced AES-256 with 9 rounds, we decrease the required number of related keys from 256 to 4 and both the data and time complexity at the cost of a smaller number of attacked rounds. Furthermore, we point out some flaw in the 9-round AES-192 attack presented at Eurocrypt 2005, show how to fix it and enhance the attack in terms of the number of related keys.

Keywords : Block Ciphers, Cryptanalysis, AES, Related-Key Rectangle Attack

1 Introduction

The Advanced Encryption Standard (AES), the successor to the Data Encryption Standard (DES), is a block cipher adopted as mandatory encryption standard by the US government. Since NIST announced that the block cipher Rijndael, designed by Daemen and Rijmen [12], was selected for the AES in 2000, it has gradually become one of the most widely used encryption algorithms in the

* This work was supported in part by the Concerted Research Action (GOA) Ambiorics 2005/11 of the Flemish Government, by the IUAP P6/26 BCRYPT of the Belgian Federal Science Policy Office and by the European Commission through the IST Programme under Contract IST 2002-507932 ECRYPT and in part by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

world. Therefore, it is important to continue studying the security of AES.

The AES algorithm is a 128-bit SP-network block cipher, which uses 128-bit, 192-bit or 256-bit keys. According to the length of the keys, the AES performs different key scheduling algorithms, different numbers of rounds, but the same round function which is made up of SubBytes (SB), ShiftRows (SR), MixColumns (MC) and AddRoundKey (ARK). These different versions of AES are referred to as AES-128, AES-192 and AES-256.

One of the most powerful known attacks on block ciphers is differential cryptanalysis [1] introduced by Biham and Shamir in 1990. It uses a differential with a non-trivial probability to retrieve subkeys for the first or last few rounds. After this attack was introduced, it has been applied effectively to many known block ciphers and various variants of this attack have been proposed such as the truncated differential attack [24], the higher order differential attack [24], the differential-linear attack [26], the impossible differential attack [3], the boomerang attack [32] and the rectangle attack [5]. Unlike differential cryptanalysis, in the boomerang and rectangle attacks [32, 5], two consecutive differentials are used, which are independent of each other, in order to retrieve subkeys for the first or last few rounds.

In 1992 and 1993, Knudsen [23] and Biham [2] independently introduced a cryptanalytic method using related keys, called the related-key attack [2], which applies differential cryptanalysis to the cipher with different, but related unknown keys. This attack is based on the key scheduling algorithm and on the encryption/decryption algorithms, hence a block cipher with a weak key scheduling algorithm may be vulnerable to this kind of attack. Several cryptanalytic results of this attack were reported in [18, 19, 10, 20].

The related-key rectangle attack introduced in [21, 16, 6] combines the rectangle and related-key attacks by applying the rectangle attack to the cipher with different, but related unknown keys: [21, 16, 6] show how to apply the rectangle attack with 2, 4, and more than 4 related keys, and show that this kind of attack can be applied to 8-round reduced AES-192 with 4 related keys [16], 9-round reduced AES-192 with 256 related keys [6], and 10-round reduced AES-256 with 256 related keys [6]. Several other articles have been published that demonstrate the power of this attack [7, 13, 27, 28].

In this paper we examine the security of AES-192 and AES-256 against a related-key rectangle attack in other related-key settings. We show that a related-key rectangle attack is applicable to 8-round reduced AES-192 with 2 related keys, 10-round reduced AES-192 with 64 or 256 related keys and 9-round reduced AES-256 with 4 related keys. Our 10-round AES-192 attack leads to the best known attack on AES-192 and our 8-round AES-192, 9-round AES-256 attacks are both better than the previously best known attacks on AES-192 with 2 related keys and AES-256 with 4 related keys in terms of the number of attacked rounds and a data or time complexity. We also demonstrate a flaw in the 9-round AES-192 attack presented at Eurocrypt 2005 [6], show how to fix it and enhance the attack in terms of the number of related keys (from 256 to 64 related keys). See Table 1 for the comparison of our results and the previous ones on AES.

Table 1. Summary of the previous attacks and our attacks on AES

Block Cipher	Type of Attack	Number of Rounds	Number of keys	Complexity Data / Time
AES-128 (10 rounds)	Imp. Diff.	5	1	$2^{29.5}$CP / 2^{31}[4]
		6	1	$2^{91.5}$CP / 2^{122}[11]
	Boomerang	6	1	2^{71}ACPC / 2^{71}[9]
	Partial Sums	6	1	$6 \cdot 2^{32}$CP / 2^{44}[14]
		7	1	$2^{128} - 2^{119}$CP / 2^{120}[14]
AES-192 (12 rounds)	Imp. Diff.	7	1	2^{92}CP / 2^{186} [31]
	Square	7	1	2^{32}CP / 2^{184} [29]
	Partial Sums	7	1	$19 \cdot 2^{32}$CP / 2^{155} [14]
		7	1	$2^{128} - 2^{119}$CP / 2^{120}[14]
		8	1	$2^{128} - 2^{119}$CP / 2^{188}[14]
	RK Imp. Diff.	7	2	2^{111}RK-CP / 2^{116} [17]
		7	32	2^{56}CP / 2^{94} [8]
		8	2	2^{88}RK-CP / 2^{183} [17]
		8	32	2^{116}CP / 2^{134} [8]
		8	32	2^{92}CP / 2^{159} [8]
		8	32	$2^{68.5}$CP / 2^{184} [8]
	RK Rectangle	8	4	$2^{86.5}$RK-CP / $2^{86.5}$[16]
		8	**2**	**2^{94}RK-CP / 2^{120}(New)**
		9†	256	2^{86}RK-CP / 2^{125}[6]
		9‡	**64**	**2^{85}RK-CP / 2^{182}(New)**
		10	**256**	**2^{125}RK-CP / 2^{182}(New)**
		10	**64**	**2^{124}RK-CP / 2^{183}(New)**
AES-256 (14 rounds)	Partial Sums	8	1	$2^{128} - 2^{119}$CP / 2^{204} [14]
		9	256	2^{85}CP / $5 \cdot 2^{224}$ [14]
	RK Rectangle	**9**	**4**	**2^{99}RK-CP / 2^{120}(New)**
		10	256	$2^{114.9}$RK-CP / $2^{171.8}$[6]
		10	**64**	**$2^{113.9}$RK-CP / $2^{172.8}$(New)**

CP: Chosen Plaintexts, ACPC: Adaptive Chosen Plaintexts and Ciphertexts
RK: Related-Key, Time: Encryption units
†: the attack with some flaw, ‡: the attack correcting the flaw in †

The outline of this paper is as follows: in Sect. 2, we give a brief description of AES and in Sect. 3, we describe a general method of the related-key rectangle attack. Sections 4 and 5 present our related-key rectangle attacks on reduced AES-192 and AES-256. Section 6 gives some comments on the previous 9-round AES-192 attack. Finally, we conclude the paper in Sect. 7.

2 Description of AES

AES encrypts data blocks of 128 bits with 128, 192 or 256-bit keys. According to the length of the keys, AES uses a different number of rounds, i.e., it has 10, 12 and 14 rounds when used with 128, 192 and 256-bit keys, respectively. The round function of AES consists of the following four basic transformations:

– SubBytes (SB) is a nonlinear byte-wise substitution that applies the same 8×8 S-box to every byte.

– ShiftRows (SR) is a cyclic shift of the *i-th* row by i bytes to the left.
– MixColumns (MC) is a matrix multiplication applied to each column.
– AddRoundKey (ARK) is an exclusive-or with the round key.

Each round function of AES applies the SB, SR, MC and ARK steps in order, but MC is omitted in the last round. Before the first round, an extra ARK step is applied. We call the key used in this step a whitening key. For more details of the above four transformations, we refer to [12].

AES uses different key scheduling algorithms according to the length of the supplied keys. The key schedule of AES-128 accepts a 128-bit key (W_0, W_1, W_2, W_3) and generates subkeys $W_4, W_5, \cdots, W_{43}$, where each W_i is a 32-bit word composed of 4 bytes in column. The subkeys are generated by the following procedure:

– For $i = s$ till $i = t$ do the following (for AES-128, $s = 4$ and $t = 43$),
  • If $i \equiv 0 \bmod s$, then $W_i = W_{i-s} \oplus \mathrm{SB}(\mathrm{RotByte}(W_{i-1})) \oplus \mathrm{Rcon}(i/s)$,
  • else $W_i = W_{i-s} \oplus W_{i-1}$,

where RotByte represents one byte rotation and Rcon denotes fixed constants depending on its input. In AES-128, the whitening key is (W_0, W_1, W_2, W_3) and the subkey of round i is $(W_{4i+4}, W_{4i+5}, W_{4i+6}, W_{4i+7})$, where $0 \leq i \leq 9$.

Similarly, the key schedules of AES-192 and AES-256 accept 192- and 256-bit keys, and generate as many subkeys as required. The key schedule of AES-192 is exactly the same as that of AES-128 except for the use of $s = 6$ and $t = 51$. The subkeys of AES-256 are derived from the following procedure:

– For $i = 8$ till $i = 59$ do the following,
  • If $i \equiv 0 \bmod 8$ then $W_i = W_{i-8} \oplus \mathrm{SB}(\mathrm{RotByte}(W_{i-1})) \oplus \mathrm{Rcon}(i/8)$,
  • If $i \equiv 4 \bmod 8$ then $W_i = W_{i-8} \oplus \mathrm{SB}(W_{i-1})$,
  • else $W_i = W_{i-8} \oplus W_{i-1}$.

In this paper a 128-bit block of AES is represented by a 4×4 byte matrix as in Fig. 1 or by $((X_0, X_1, X_2, X_3), (X_4, X_5, X_6, X_7), (X_8, X_9, X_{10}, X_{11}), (X_{12}, X_{13}, X_{14}, X_{15}))$.

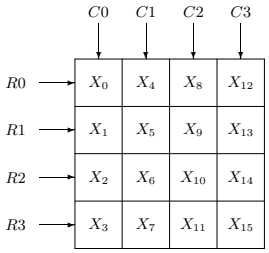

Fig. 1. Byte coordinate of a 128-bit block of AES
(Ri: Row i, Ci: Column i, X_i: Byte i)

3 The Related-Key Rectangle Attack

The related-key rectangle attack is based on two consecutive related-key differentials with relatively high probabilities which are independent of each other: the underlying cipher $E : \{0,1\}^n \times \{0,1\}^k \to \{0,1\}^n$ is treated as a cascade of two sub-ciphers $E = E^1 \circ E^0$, where $\{0,1\}^k$ and $\{0,1\}^n$ are the key space and the plaintext/ciphertext space, respectively. We assume that for E^0 there exists a related-key differential $\alpha \to \beta$ with probability p and for E^1 there exists a related-key differential $\gamma \to \delta$ with probability q. More precisely,

$$p = Pr[E^0_K(X) \oplus E^0_{K^*}(X \oplus \alpha) = \beta] = Pr[E^0_{K'}(X) \oplus E^0_{K'^*}(X \oplus \alpha) = \beta],$$

$$q = Pr[E^1_K(X) \oplus E^1_{K'}(X \oplus \gamma) = \delta] = Pr[E^1_{K^*}(X) \oplus E^1_{K'^*}(X \oplus \gamma) = \delta],$$

where $K^* = K \oplus \Delta K$, $K' = K \oplus \Delta K'$ and $K'^* = K \oplus \Delta K \oplus \Delta K'$, i.e., $K \oplus K^* = K' \oplus K'^* = \Delta K$ and $K \oplus K' = K^* \oplus K'^* = \Delta K'$ for known key differences ΔK and $\Delta K'$. Then, these consecutive related-key differentials can be used efficiently to the following related-key rectangle distinguisher:

- Choose two random n-bit plaintexts P and P' and compute two other plaintexts $P^* = P \oplus \alpha$ and $P'^* = P' \oplus \alpha$.
- With a chosen plaintext attack scenario, obtain the corresponding ciphertexts $C = E_K(P)$, $C^* = E_{K^*}(P^*)$, $C' = E_{K'}(P')$ and $C'^* = E_{K'^*}(P'^*)$.
- Check if $C \oplus C' = C^* \oplus C'^* = \delta$.

The probability that the ciphertext quartet $(C, C^*, C'C'^*)$ satisfies the last δ test is computed as follows: let X, X^*, X' and X'^* denote the encrypted values of P, P^*, P' and P'^* under E^0 with K, K^*, K' and K'^*, respectively. Then, the probability that $X \oplus X^* = X' \oplus X'^* = \beta$ is about p^2 by the related-key differential for E^0. In the above process, we randomly choose two plaintexts P and P', so we expect $X \oplus X' = \gamma$ with probability 2^{-n}. Once the two above events occur, $X^* \oplus X'^* = (X \oplus X^*) \oplus (X' \oplus X'^*) \oplus (X \oplus X') = \beta \oplus \beta \oplus \gamma = \gamma$ with probability 1. Since the probability of the related-key differential $\gamma \to \delta$ for E^1 is q, $X \oplus X' = X^* \oplus X'^* = \gamma$ goes to $C \oplus C' = C^* \oplus C'^* = \delta$ with a probability of about q^2. Therefore, the total probability that the last δ test in the above process is satisfied is about

$$\sum_{\beta,\gamma} p^2 \cdot 2^{-n} \cdot q^2 = \widehat{p}^2 \cdot \widehat{q}^2 \cdot 2^{-n}, \text{where } \widehat{p} = \sqrt{\sum_{\beta} p^2} \text{ and } \widehat{q} = \sqrt{\sum_{\gamma} q^2}\ .$$

On the other hand, for a random cipher, the δ test holds with probability 2^{-2n} and thus if the above probability is larger than 2^{-2n} for any 4-tuple $(\alpha, \delta, \Delta K, \Delta K')$, i.e., if $\widehat{p} \cdot \widehat{q} > 2^{-n/2}$, the related-key rectangle distinguisher can be used to distinguish E from a random cipher. Similarly, two consecutive related-key truncated differentials can be used to form a related-key rectangle distinguisher.

According to the condition of the key differences ΔK and $\Delta K'$, the above related-key rectangle distinguisher is used in different ways. If $\Delta K \neq 0$ and

$\Delta K' = 0$, then the distinguisher works with two related keys; a related-key differential for E^0 and a regular (non-related-key) differential for E^1. If $\Delta K = 0$ and $\Delta K' \neq 0$, then the distinguisher also works with two related keys; however, a regular differential for E^0 and a related-key differential for E^1 are used. If $\Delta K \neq 0$, $\Delta K' \neq 0$ and $\Delta K \neq \Delta K'$, then the distinguisher works with four related keys, in which related-key differentials for both E^0 and E^1 are used. Further, more than four related keys can be used in the related-key rectangle distinguisher as in [6, 7]; the basic idea is the same as that of the distinguisher with two or four related keys.

4 Related-Key Rectangle Attack on 10-Round AES-192

This section shows how to exploit the related-key rectangle attack to devise key recovery attacks on 10-round AES-192 with 64 or 256 related keys. We first focus on 10-round AES-192 with 256 related keys.

Denote the 10 rounds of AES-192 by $E = E^f \circ E^1 \circ E^0 \circ E^b$, where E^b is round 0 including the whitening key addition step and excluding the key addition step of round 0, E^0 is rounds 1-4 including the key addition step of round 0, E^1 is rounds 5-8 and E^f is round 9. In our 10-round AES-192 attack, we use a related-key truncated differential for E^0 depicted in Fig. 2 and another related-key truncated differential for E^1 depicted in Fig. 3. After we convert these related-key truncated differentials for E^0 and E^1 into a related-key rectangle distinguisher for $E^1 \circ E^0$, we apply it to recover some portions of the keys in E^b and E^f. Before describing our attack, we define some notation which is used in our attacks on AES.

- $K_w, K_w^*, K_w', K_w'^*$: whitening keys generated from master keys K, K^*, K',K'^*, respectively.
- $K_i, K_i^*, K_i', K_i'^*$: subkeys of round i generated from K, K^*, K', K'^*, respectively.
- P, P^*, P', P'^*: plaintexts encrypted under K, K^*, K', K'^*, respectively.
- $I_i, I_i^*, I_i', I_i'^*$: input values to round i caused by plaintexts P, P^*, P', P'^* under K, K^*, K', K'^*, respectively.
- a: a fixed nonzero byte value.
- b, c: output differences of S-box for the fixed nonzero input difference a.
- $*$: a variable and unknown byte.

4.1 8-Round Related-Key Rectangle Distinguisher

Our related-key truncated differentials depicted in Figs. 2 and 3 exploit the slow difference propagation of the key schedule of AES-192, which makes it possible that 3-round key differences $\Delta K_0 || \Delta K_1 || \Delta K_2$ and $\Delta K_5' || \Delta K_6' || \Delta K_7'$ satisfy $HW_b(\Delta K_0) = HW_b(\Delta K_5') = 2$, $HW_b(\Delta K_1) = HW_b(\Delta K_6') = 0$ and $HW_b(\Delta K_2) = HW_b(\Delta K_7') = 1$, where $HW_b(X)$ is the byte Hamming weight of X. Using these key differences with small byte Hamming weights we make $\Delta I_1 =$

$\Delta I'_6 = 0$ in our related-key truncated differentials which induce $HW_b(\Delta I_3) = HW_b(\Delta I'_8) = 1$ (see Figs. 2 and 3 for ΔK_i, $\Delta K'_i$, ΔI_i and $\Delta I'_i$) and we add one or two more rounds to get longer related-key differentials. Note that our related-key truncated differential for E^0 is the same as the one for rounds 0-3 (including the whitening key addition step) used in [16].

In order to convert the two 4-round related-key truncated differentials into an 8-round related-key rectangle distinguisher, we first make the following Assumptions 1, 2 and 3.

Assumption 1. The key quartet (K, K^*, K', K'^*) is related as follows;

$$K \oplus K^* = K' \oplus K'^* = \Delta K,\ K \oplus K' = K^* \oplus K'^* = \Delta K' .$$

Assumption 2. A plaintext quartet (P, P^*, P', P'^*) is related as follows;

$$P \oplus P^*,\ P' \oplus P'^* \in \Delta P .$$

Assumption 3. $E^b_K(P) \oplus E^b_{K^*}(P^*) = E^b_{K'}(P') \oplus E^b_{K'^*}(P'^*) = \Delta K_0$.

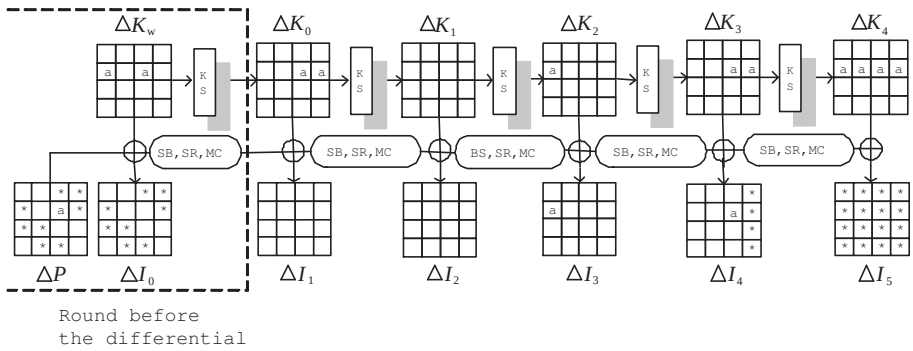

Fig. 2. The first related-key truncated differential for rounds 1-4 including the key addition step of round 0 (E^0), and the preceding differential for round 0 including the whitening key addition step and excluding the key addition step of round 0 (E^b)

Note that ΔK is the same as the first six columns of $\Delta K_w || \Delta K_0$ in Fig. 2. As stated in our notation, $I_5 = E^0_K(E^b_K(P))$, $I^*_5 = E^0_{K^*}(E^b_{K^*}(P^*))$, $I'_5 = E^0_{K'}(E^b_{K'}(P'))$ and $I'^*_5 = E^0_{K'^*}(E^b_{K'^*}(P'^*))$. By the related-key truncated differential for E^0, $I_5 \oplus I^*_5$ is equal to $I'_5 \oplus I'^*_5$ with a probability of about $(2^{-32} \cdot 2^{-7})^2 \cdot (2^7 - 2) \cdot 2^{32} + (2^{-32} \cdot 2^{-6})^2 \cdot 2^{32} \approx 2^{-39}$ (this probability has been used for the 8-round AES-192 attack presented in [16]). It follows from counting over all the differentials that can be generated by the active S-box with input difference a and the

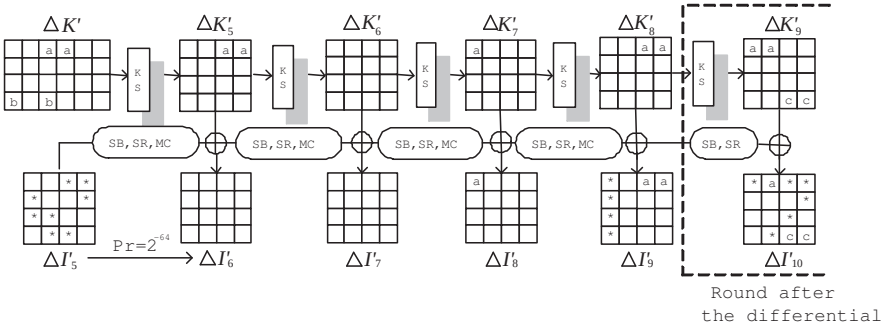

Fig. 3. The second related key truncated differential for rounds 5-8 (E^1) and the following differential for round 9 (E^f)

other four active S-boxes in round 4. Since ShiftRows and MixColumns are linear layers, they can be ignored in round 4 when computing the probability (see Fig. 2). Moreover, the probability that $I_5 \oplus I'_5$, $I^*_5 \oplus I'^*_5 \in \Delta I'_5$ is about 2^{-64} under the condition $I_5 \oplus I^*_5 = I'_5 \oplus I'^*_5$ (see Fig. 3 for $\Delta I'_5$). Hence the probability that $I_5 \oplus I'_5,\ I^*_5 \oplus I'^*_5 \in \Delta I'_5$ is about $2^{-39} \cdot 2^{-64} = 2^{-103}$. Since $e_K(I_5) \oplus e_{K'}(I'_5) = 0$ with probability 2^{-64} and $e_{K^*}(I^*_5) \oplus e_{K'^*}(I'^*_5) = 0$ with a probability of about 2^{-64} under the condition $I_5 \oplus I'_5,\ I^*_5 \oplus I'^*_5 \in \Delta I'_5$, where e is the encryption for round 5,

$$E^1_K(I_5) \oplus E^1_{K'}(I'_5),\ E^1_{K^*}(I^*_5) \oplus E^1_{K'^*}(I'^*_5) \in \Delta I'_9$$

with a probability of 2^{-231} (see Fig. 3 for $\Delta I'_9$). However, the same statement can be applied to a random cipher with probability $(2^{-128} \cdot (2^7 - 1))^2 \approx 2^{-242}$, since the number of elements in $\Delta I'_9$ is $2^7 - 1$. The first column of $\Delta I'_9$ is

$$\mathcal{B} = \{\mathrm{MC}(y, 0, 0, 0) \mid y = \mathrm{BS}(x) \oplus \mathrm{BS}(x \oplus a),\ x = 0, 1, 2, \cdots, 255\}. \quad (1)$$

4.2 Key Recovery Attack on 10-Round AES-192 with 256 Related Keys

In order to produce the round-key differences depicted in Fig. 3, the 8-bit difference a should satisfy the 8-bit difference b after S-box during the key generation for the third column of $\Delta K'_3$. Given the 8-bit difference a there are 127 possible candidates for the b difference, hence the attack starts by gathering all possible key quartets $(K, K^*, \widetilde{K}', \widetilde{K}'^*)$ of which one satisfies the desired key condition. Note that the keys $K^* = K \oplus \Delta K$, $\widetilde{K}' = K \oplus \Delta\widetilde{K}'$ and $\widetilde{K}'^* = K \oplus \Delta K \oplus \Delta\widetilde{K}'$ where ΔK is fixed as ΔK_w and the first two columns of ΔK_0 in Fig. 2 and $\Delta\widetilde{K}'$ is one of the 127 possible differences; bytes 8 and 12 are both a, bytes 3 and 11 are both b' and other bytes are all zeros, where b' is one of the 127 possible candidates for the b difference. So the total number of required related keys is

256. We apply the related-key rectangle attack to 10-round AES-192 for each key quartet. During this procedure, we stop our attack when we have found a key quartet $(K, K^*, \widetilde{K}', \widetilde{K}'^*)$ that satisfies the desired key condition $b' = b$, i.e., $\Delta\widetilde{K}' = \Delta K'$, $(K, K^*, \widetilde{K}', \widetilde{K}'^*) = (K, K^*, K', K'^*)$.

The aim of our attack is to recover bytes 1, 2, 6, 7, 8, 11, 12, 13 of the whitening key quartet $(K_w, K_w^*, K_w', K_w'^*)$ and bytes 0, 7, 8, 10, 12, 13 of the subkey quartet $(K_9, K_9^*, K_9', K_9'^*)$, for which the byte positions are marked as $*$ on ΔP and $\Delta I_{10}'$ depicted in Fig. 2 and Fig. 3. This attack distinguishes a right key quartet from wrong ones by analyzing enough plaintext quartets with each guessed key quartet. In this attack, we need 2^{64} guesses for the whitening key quartet and 2^{72} guesses for the subkey quartet in round 9, since bytes 0, 7, 8, 10, 12, 13 of ΔK_9 are $d,\ 0,\ d,\ e,\ 0,\ f$, respectively, where d, e and f are unknown 8-bit values (note that bytes 0, 7, 8, 10, 12, 13 of $\Delta K_9'$ are fixed by $a,\ 0,\ 0,\ 0,\ 0,\ 0$, respectively). Thus, taking into account the guessing of candidates for the difference b, we need about 2^{143} key guesses in total (in our attack it can be reduced by a factor of two on average).

The attack algorithm goes as follows:

1. Choose 2^{54} structures $S_1,\ S_2, \cdots,\ S_{2^{54}}$ of 2^{64} plaintexts each, where in each structure the 64 bits of bytes 0, 3, 4, 5, 9, 10, 14, 15 are fixed. With a chosen plaintext attack scenario, obtain all their corresponding ciphertexts under the key K. (Step 1 takes 2^{118} chosen plaintexts and about 2^{118} encryptions. Note that n encrpytions mean n 10-round AES-192 encryptions.)

2. Compute 2^{54} structures $S_1^*,\ S_2^*, \cdots,\ S_{2^{54}}^*$ of 2^{64} plaintexts each by XORing the plaintexts in $S_1,\ S_2, \cdots,\ S_{2^{54}}$ with a 128-bit value M of which byte 9 is a and all the other bytes are 0. With a chosen plaintext attack scenario, obtain all their corresponding ciphertexts under the key K^*, where $K^* = K \oplus \Delta K$. (Similarly, Step 2 takes 2^{118} chosen plaintexts and about 2^{118} encryptions.)

3. Guess a candidate for the difference b and compute $\Delta\widetilde{K}'$. For the key difference $\Delta\widetilde{K}'$, do the following:

3.1 Choose 2^{54} structures $S_1',\ S_2', \cdots,\ S_{2^{54}}'$ of 2^{64} plaintexts each, where in each structure the 64 bits of bytes 0, 3, 4, 5, 9, 10, 14, 15 are fixed. With a chosen plaintext attack scenario, obtain all their corresponding ciphertexts under the key $\widetilde{K}'$, where $\widetilde{K}' = K \oplus \Delta\widetilde{K}'$. (For each guess of $\Delta\widetilde{K}'$, Step 3.1 takes 2^{118} chosen plaintexts and about 2^{118} encryptions.)

3.2 Compute 2^{54} structures $S_1'^*,\ S_2'^*, \cdots,\ S_{2^{54}}'^*$ of 2^{64} plaintexts each by XORing the plaintexts in $S_1',\ S_2', \cdots,\ S_{2^{54}}'$ with M. With a chosen plaintext attack scenario, obtain all their corresponding ciphertexts under the key $\widetilde{K}'^*$, where $\widetilde{K}'^* = K \oplus \Delta K \oplus \Delta\widetilde{K}'$. Go to Step 4. (For each guess of $\Delta\widetilde{K}'$, this step also takes 2^{118} chosen plaintexts and about 2^{118} encryptions.)

4. Guess a 64-bit subkey k_w in the position of bytes 1, 2, 6, 7, 8, 11, 12, 13 of the whitening key and compute $k_w^* = k_w \oplus \Delta k_w$, $k_w' = k_w \oplus \Delta\widetilde{k}_w'$, $k_w'^* = k_w \oplus \Delta k_w \oplus \Delta\widetilde{k}_w'$, where Δk_w and $\Delta\widetilde{k}_w'$ are the fixed 64-bit key differences in the position of bytes 1, 2, 6, 7, 8, 11, 12, 13 of ΔK_w (depicted in Fig. 2) and $\Delta\widetilde{K}_w'$, respectively. For the subkey quartet $(k_w, k_w^*, k_w', k_w'^*)$, do the following:

4.1 Partially encrypt each plaintext P_{i,l_0} in S_i through E^b under k_w, $i = 1, 2, \cdots, 2^{54}$, $l_0 = 1, 2, \cdots, 2^{64}$. We denote the partially encrypted value by x_{i,l_0}. Partially decrypt each $x_{i,l_0} \oplus \Delta K_{0,R}$ through E^b under k_w^*, and find the corresponding plaintext in S_i^*, denoted P_{i,l_0}^*. We denote the corresponding ciphertexts of P_{i,l_0} and P_{i,l_0}^* by C_{i,l_0} and C_{i,l_0}^*, respectively. (For each guess of 64-bit k_w, Step 4.1 takes about $2^{64+1} \cdot (8/16) \cdot (1/10) = 2^{60.7}$ encryptions. Note that this step is independent of $\Delta\widetilde{K}'$ in Step 3, so there is no need to run this step for every iteration of Step 3.)

4.2 Partially encrypt each plaintext P'_{j,l_1} in S'_j through E^b under k'_w, $j = 1, 2, \cdots, 2^{54}$, $l_1 = 1, 2, \cdots, 2^{64}$. We denote the partially encrypted value by x'_{j,l_1}. Partially decrypt each $x'_{j,l_1} \oplus \Delta K_{0,R}$ through E^b under k'^*_w, and find the corresponding plaintext in S'^*_j, denoted P'^*_{j,l_1}. We denote the corresponding ciphertexts of P'_{j,l_1} and P'^*_{j,l_1} by C'_{j,l_1} and C'^*_{j,l_1}, respectively. (For each guess of 71-bit $(k_w, \Delta\widetilde{K}')$, Step 4.2 takes about $2^{64+1} \cdot (8/16) \cdot (1/10) = 2^{60.7}$ encryptions.)

4.3 Insert $C_{i,l_0}||C^*_{i,l_0}$ in a hash table (indexed by bytes 1, 2, 3, 4, 5, 6, 9, 14) and then check that $(C_{i,l_0}||C^*_{i,l_0}) \oplus (C'_{j,l_1}||C'^*_{j,l_1}) \in \Delta I'_{10}(1)||\Delta I'_{10}(2)$ for all i, j, l_0 and l_1, where $\Delta I'_{10}(1) = \{((*, 0, 0, 0), (a, 0, 0, *), (b_1, 0, *, b_2), (b_3, *, 0, b_2))\}$, $\Delta I'_{10}(2) = \{((*, 0, 0, 0), (a, 0, 0, *), (b_4, 0, *, b_2), (b_5, *, 0, b_2))\}$, $*$ is any 8-bit value, and b_i is one of the output differences caused by the input difference a to the S-box. Note that $\Delta I'_{10}(1)$ and $\Delta I'_{10}(2)$ are both candidates for $\Delta I'_{10}$, i.e., b_2 is a candidate for c (see Fig. 3). Keep in a table all the ciphertext quartets $(C_{i,l_0}, C'_{j,l_1}, C^*_{i,l_0}, C'^*_{j,l_1})$ passing the both tests and go to Step 5 with this table. Since $\Delta I'_{10}(1)$ in the first test has 2^{53} out of 2^{128} values and $\Delta I'_{10}(2)$ in the second test has 2^{46} out of 2^{128} values, the expected number of quartets kept in the table is about $2^{(54+64)\cdot 2} \cdot 2^{-128+53} \cdot 2^{-128+46} = 2^{79}$. (For each guess of 71-bit $(k_w, \Delta\widetilde{K}')$, Step 4.3 takes about 2^{119} memory accesses that are equivalent to approximately 2^{112} encryptions according to the implementations of NESSIE primitives [30].)

5. Guess an 8-bit subkey $k_{9,v}$ in the position of byte 12 in round 9 and set $k^*_{9,v} = k'_{9,v} = k'^*_{9,v} = k_{9,v}$. For the 8-bit subkey quartet $(k_{9,v}, k^*_{9,v}, k'_{9,v}, k'^*_{9,v})$, do the following:

5.1 For all the remaining ciphertext quartets $(C_{i,l_0}, C'_{j,l_1}, C^*_{i,l_0}, C'^*_{j,l_1})$, partially decrypt C_{i,l_0} and C'_{j,l_1} under $k_{9,v}$ and $k'_{9,v}$ through E^f, respectively. If the partially decrypted pairs do not have the difference a, then discard the corresponding ciphertext quartets. Since it has approximately a 7-bit filtering, the number of remaining quartets after this step is about 2^{72}. (The partial decryptions can be done after the remaining ciphertext quartets have been sorted by byte 12 of (C_{i,l_0}, C'_{j,l_1}) or this step can use a pre-computed table, so Step 5.1 takes a relatively small time complexity.)

5.2 For all the remaining ciphertext quartets $(C_{i,l_0}, C'_{j,l_1}, C^*_{i,l_0}, C'^*_{j,l_1})$, partially decrypt C^*_{i,l_0} and C'^*_{j,l_1} under $k^*_{9,v}$ and $k'^*_{9,v}$ through E^f, respectively. If the partially decrypted pairs do not have the difference a, discard the

corresponding ciphertext quartets and then go to Step 6. It also imposes approximately a 7-bit filtering, hence the number of remaining quartets after this step is about 2^{65}. (Similarly, Step 5.2 can be performed efficiently.)

6. Guess an 8-bit subkey $k_{9,w}$ in the position of byte 8 in round 9 and set $k'_{9,w} = k_{9,w}$. For the 8-bit subkey pair ($k_{9,w}$,$k'_{9,w}$), do the following:

6.1 For all the remaining ciphertext quartets $(C_{i,l_0}, C'_{j,l_1}, C^*_{i,l_0}, C'^*_{j,l_1})$, partially decrypt C_{i,l_0} and C'_{j,l_1} under $k_{9,w}$ and $k'_{9,w}$ through E^f, respectively. If the partially decrypted pairs do not have the difference a, then discard the corresponding ciphertext quartets. Since this imposes approximately a 7-bit filtering, the number of remaining quartets after this step is about 2^{58}.

6.2 Guess an 8-bit value d to form an 8-bit subkey pair ($k^*_{9,w} = k_{9,w} \oplus d$, $k'^*_{9,w} = k_{9,w} \oplus d$) in the position of byte 8 in round 9. For the 8-bit subkey pair ($k^*_{9,w}$,$k'^*_{9,w}$), do the following:

6.2.1 For all the remaining ciphertext quartets $(C_{i,l_0}, C'_{j,l_1}, C^*_{i,l_0}, C'^*_{j,l_1})$, partially decrypt C^*_{i,l_0} and C'^*_{j,l_1} under $k^*_{9,w}$ and $k'^*_{9,w}$ through E^f, respectively. If the partially decrypted pairs do not have the difference a, discard the corresponding ciphertext quartets and then go to Step 7. It also induces approximately a 7-bit filtering, hence the number of remaining quartets after this step is about 2^{51}. (Similarly, Step 6 can be performed efficiently.)

7. Guess a 32-bit subkey $k_{9,y}$ in the position of bytes 0, 7, 10, 13 in round 9 and compute $k'_{9,y} = k_{9,y} \oplus (a, 0, 0, 0)$. For the 32-bit subkey pair $(k_{9,y}, k'_{9,y})$, do the following:

7.1 For all the remaining ciphertext quartets $(C_{i,l_0}, C'_{j,l_1}, C^*_{i,l_0}, C'^*_{j,l_1})$, partially decrypt C_{i,l_0} and C'_{j,l_1} under $k_{9,y}$ and $k'_{9,y}$ through E^f, respectively. If the differences of the partially decrypted pairs are not in $\mathcal{B}$ (see Eq. (1)), then discard the corresponding ciphertext quartets. Since $\mathcal{B}$ has $2^7 - 1$ out of 2^{32} values, the remaining quartets after this step is about 2^{26}. (For each guess of 127-bit $(k_{9,y}, d, k_{9,w}, k_{9,v}, k_w, \Delta\widetilde{K}')$, Step 7.1 takes $2^{51+1} \cdot (4/16) \cdot (1/10) = 2^{46.7}$ encryptions.)

7.2 Guess two 8-bit values e, f to form a 32-bit subkey pair ($k^*_{9,y} = k_{9,y} \oplus (d, 0, e, f)$, $k'^*_{9,y} = k_{9,y} \oplus (d \oplus a, 0, e, f)$) in the position of bytes 0, 7, 10, 13 in round 9. For the 32-bit subkey pair ($k^*_{9,y}$,$k'^*_{9,y}$), do the following:

7.2.1 For all the remaining ciphertext quartets $(C_{i,l_0}, C'_{j,l_1}, C^*_{i,l_0}, C'^*_{j,l_1})$, partially decrypt C^*_{i,l_0} and C'^*_{j,l_1} under $k^*_{9,y}$ and $k'^*_{9,y}$ through E^f, respectively. If the differences of the partially decrypted pairs are not in $\mathcal{B}$, discard the corresponding ciphertext quartets and then go to Step 8. This also induces approximately about a 25-bit filtering, hence the number of remaining quartets after this step is about 2 for each wrong key guess. (For each guess of 143-bit $(e, f, k_{9,y}, d, k_{9,w}, k_{9,v}, k_w, \Delta\widetilde{K}')$, this step takes $2^{26+1} \cdot (4/16) \cdot (1/10) = 2^{21.7}$ encryptions.)

8. For the remaining ciphertext quartets $(C_{i,l_0}, C'_{j,l_1}, C^*_{i,l_0}, C'^*_{j,l_1})$, classify the quartets according to the differences of C_{i,l_0} and C'_{j,l_1} by byte 11. Discard all the ciphertext quartets except for the group with the largest number of quartets and then go to Step 9. Since this results in approximately a 7-bit filtering for each pair of quartets, the remaining quartets after this step is expected to be about 2^{-6} for each wrong key guess. (It takes a relatively small time complexity.)

9. If there are more than 16 quartets in the table, then output the guessed subkey quartet as the right one. Otherwise, run the above steps with another guess for the subkey quartet, i.e., $(e, f, k_{9,y}, d,\ k_{9,w}, k_{9,v}, k_w, \Delta\widetilde{K}')$.

About 2^{125} chosen plaintexts in Steps 1, 2 and 3 are encrypted on average, hence the data complexity of this attack is about 2^{125} related-key chosen plaintexts and the time complexity of Steps 1, 2 and 3 is about 2^{125} encryptions. Step 4 runs about 2^{70} times, so the time complexity of Step 4.2 is about $2^{60.7+70} = 2^{130.7}$ encryptions (it can be improved by a factor of about 2^4 by using a pre-computed table[1]) and the time complexity of Step 4.3 is about $2^{112+70} = 2^{182}$ encryptions. As stated above, Steps 5, 6 and 8 take relatively small time complexities compared to other steps.

The time complexity for Step 7 depends on how many times this step runs, which can be measured by the number of guessed subkeys (including d, e and f). Since Steps 7.1 and 7.2 run in this attack 2^{126} and 2^{142} times on average, these steps take $2^{172.7}$ and $2^{163.7}$ encryptions, respectively. However, the time complexities of these steps can be improved by using a divide and conquer technique. In Step 7.1, two of the four bytes of the remaining ciphertext quartets are first decrypted (these partial decryptions can be performed after the remaining ciphertext quartets are sorted by these two bytes) and discard the ciphertext quartets of which the decrypted two bytes do not have a difference in $\mathcal{B}$ with respect to the two-byte position, and then do this test with other two bytes of the remaining ciphertext quartets byte by byte. With this divide and conquer technique, we can also run Step 7.2. This method allows Steps 7.1 and 7.2 to decrease their time complexities down to about $2^{135.7}$ and $2^{146.7}$ encryptions, respectively.

We can calculate the success rate of the attack by using the Poisson distribution. Since the expected number of remaining quartets for each wrong subkey quartet is 2^{-6}, the probability that the number of remaining quartets for each wrong subkey quartet is larger than 16 is 2^{-150} by the Poisson distribution, $X \sim \mathrm{Poi}(\lambda = 2^{-6})$, $Pr_X[X > 16] \approx 2^{-150}$. It follows that the probability that the attack outputs a wrong subkey quartet is quite low, since the total number of guessed wrong subkey quartets is about 2^{142}. On the other hand, the expected number of remaining quartets for the right subkey quartet is about $2^5 = 2^{236} \cdot 2^{-231}$ due to our 8-round related-key rectangle distinguisher. Thus,

[1] Before running this attack, we can pre-compute a table which keeps 2^{64} input pairs (I_0, I_0^*) to round 0, where $I_0^* = \mathrm{BS}^{-1}(\mathrm{SR}^{-1}(\mathrm{MC}^{-1}(\mathrm{MC}(\mathrm{SR}(\mathrm{BS}(I_0))) \oplus \Delta K_{0,R})))$. If Step 4.1 has access to this table for each guessed subkey (k_w, k_w^*), it can find plaintext pairs $(P_{i,l}, P_{i,l}^*)$ by XORing (k_w, k_w^*) with (I_0, I_0^*).

the probability that the number of remaining quartets for the right key quartet is larger than 16 is 0.99 by the Poisson distribution, $Y \sim \mathrm{Poi}(\lambda = 2^5)$, $Pr_Y[Y > 16] \approx 0.99$.

Therefore, this attack works with a data complexity of about 2^{125} related-key chosen plaintexts and with a time complexity of about 2^{182} encryptions and with a success rate of 0.99.

4.3 Reducing the Number of Related Keys from 256 to 64

If we take more delicate related keys in our attack, we can reduce the number of related keys from 256 to 64 (note that the basic idea of this method has been introduced in [8]). The following 64 related keys can be used in our attack:

- 16 key candidates K^i $(i = 0, 1, \cdots, 15)$ for the key K such that all the bytes of the 16 K^i have the same values in each byte position except that byte 3 is the same as byte 11 in each K^i, say s_i, but $s_0,\ s_1, \cdots,\ s_{15}$ are all pairwise distinct.
- 16 key candidates K^{*i} for the key K^* such that bytes 1 and 9 of $K^i \oplus K^{*i}$ are both a, and the other bytes of $K^i \oplus K^{*i}$ are all 0.
- 16 key candidates K'^j $(j = 0, 1, \cdots, 15)$ for the key K' such that all the bytes of the 16 K'^j are the same as those of K^i for some i except that byte 3 is the same as byte 11 in each K'^j, say t_j, but $t_0,\ t_1, \cdots,\ t_{15}$ are all pairwise distinct, and bytes 8 and 12 of $K'^j \oplus K^i$ are both a.
- 16 key candidates K'^{*j} for the key K'^* such that bytes 1 and 9 of $K'^{*j} \oplus K'^j$ are both a, and the other bytes of $K'^{*j} \oplus K'^j$ are all 0.

Using these delicately chosen key relationships, we can make 256 key quartets $(K^i, K^{*i}, K'^j, K'^{*j})$ of which one is expected to satisfy the desired key condition, $K^i \oplus K^{*i} = K'^j \oplus K'^{*j} = \Delta K$ and $K^i \oplus K'^j = K^{*i} \oplus K'^{*j} = \Delta K'$ (note that bytes 3 and 11 of $K^i \oplus K'^j$ and $K^{*i} \oplus K'^{*j}$ are both $s_i \oplus t_j$ of which one is expected to be b).

If the above 64 related keys are used in our attack algorithm, the attack works with a data complexity of 2^{124} related-key chosen plaintexts (due to the fact that the attack takes 2^{118} chosen plaintext queries for each key) and with a time complexity of 2^{183} encryptions (due to the fact that Step 4.3 is iterated 2^7 times on average by the 256 key quartets).

5 Related-Key Rectangle Attacks on 8-Round AES-192 and 9-Round AES-256

Similarly, we can construct related-key rectangle attacks on 8-round AES-192 with two related keys ($\Delta K \neq 0$ and $\Delta K' = 0$) and on 9-round AES-256 with four related keys ($\Delta K \neq 0$, $\Delta K' \neq 0$ and $\Delta K \neq \Delta K'$).

The attack on 8-round AES-192 with two related keys recovers bytes 1, 2, 6, 7, 8, 11, 12, 13 of the whitening key pair (K_w, K_w^*) and bytes 3, 6, 9, 12 of the subkey pair (K_7, K_7^*) with a data complexity of about 2^{94} related-key chosen

plaintexts, a time complexity of about 2^{120} encryptions and a success rate of 0.9. See Figs. 2 and 4 for a schematic description of our 8-round AES-192 attack (note that the related-key truncated differential in Fig. 2 is used for E^b and E^0 in this attack).

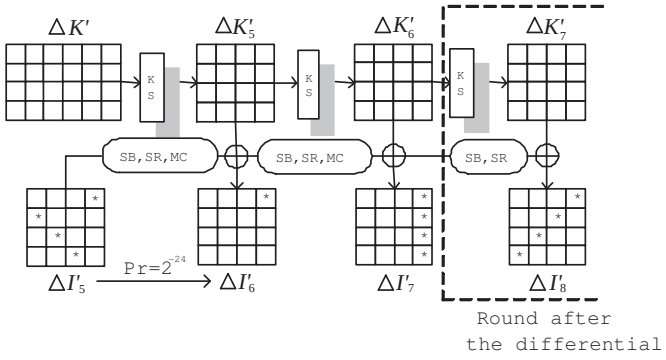

Fig. 4. The second truncated differential for rounds 5-6 (E^1) and the following differential for round 7 (E^f)

The attack on 9-round AES-256 with four related keys recovers bytes 1, 2, 6, 7, 8, 11, 12, 13 of the whitening key quartet $(K_w, K_w^*, K_w', K_w'^*)$ and bytes 0, 4, 8, 12 of the subkey quartet $(K_8, K_8^*, K_8', K_8'^*)$ with a data complexity of about 2^{99} related-key chosen plaintexts, a time complexity of about 2^{120} encryptions and a success rate of 0.9. See Figs. 5 and 6 for a schematic description of our 9-round AES-256 attack.

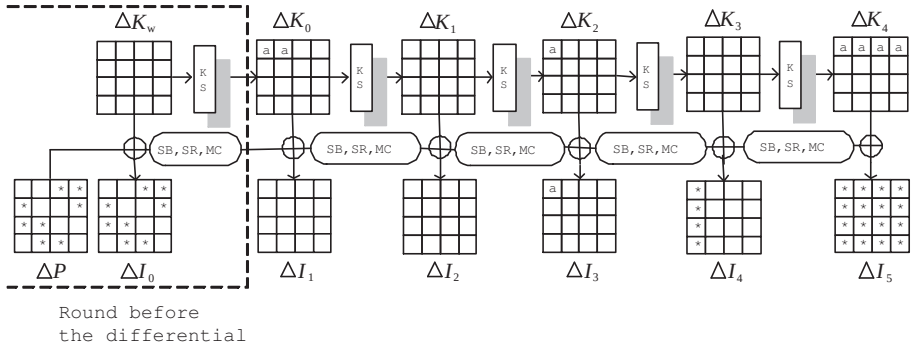

Fig. 5. The first related-key truncated differential for rounds 1-4 including the key addition step of round 0 (E^0), and the preceding differential for round 0 including the whitening key addition step and excluding the key addition step of round 0 (E^b)

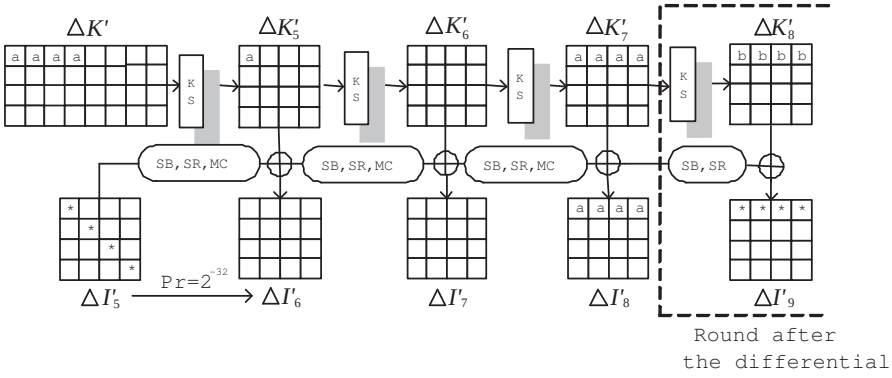

Fig. 6. The second related-key truncated differential for rounds 5-7 (E^1) and the following differential for round 8 (E^f)

6 Comments on the 9-Round AES-192 Attack Presented at Eurocrypt 2005

At Eurocrypt 2005, Biham, Dunkelman and Keller [6] presented a 9-round AES-192 attack which requires 256 related keys, a data complexity of 2^{86} related-key chosen plaintexts and a time complexity of 2^{125} encryptions. However, we have observed that there is some flaw in the key guessing step in their attack. In order to complete their attack, we need to guess in addition 56 bits of the subkey in the last round, hence the attack requires a larger time complexity of $2^{181} = 2^{125} \cdot 2^{56}$ rather than 2^{125} encryptions.

Moreover, their attack can be mounted based on 64 related keys as in our 10-round AES-192 attack. Similarly, it allows the 9-round AES-192 attack to work with a smaller data complexity, but a larger time complexity than the original ones; in our observation their attack works with 64 related keys, a data complexity of 2^{85} related-key chosen plaintexts and a time complexity of 2^{182}. This method (for reducing the number of related keys) can also be used in the 10-round AES-256 attack presented at Eurocrypt 2005. See Table 1 for the attack complexity.

7 Conclusion

In this paper we have presented related-key rectangle attacks on 8-round AES-192 with 2 related keys, 10-round AES-192 with 64 or 256 related keys and 9-round AES-256 with 4 related keys, which are faster than exhaustive key search. All our attacks have been designed based on the key scheduling algorithms of AES-192 and AES-256 which have relatively slow difference propagations.

Our 10-round AES-192 attack leads to the best known attack on AES-192 and our 8-round AES-192, 9-round AES-256 attacks are both better than previously

best known attacks on AES-192 with 2 related keys and AES-256 with 4 related keys in terms of the number of attacked rounds and the data or time complexity. It should be clear, however, that none of these attacks presents a realistic threat to the security of the AES.

Acknowledgements We thank Orr Dunkelman and Nathan Keller for their helpful comments.

References

1. E. Biham and A. Shamir, *Differential Cryptanalysis of DES-like Cryptosystems*, Advances in Cryptology – Proceedings of CRYPTO 1990, LNCS 537, pp. 2-21, Springer-Verlag, 1990.
2. E. Biham, *New Types of Cryptanalytic Attacks Using Related Keys*, Journal of Cryptology, Vol. 7, No. 4, pp. 229-246, 1994.
3. E. Biham, A. Biryukov and A. Shamir, *Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials*, Journal of Cryptology, Vol. 18, No. 4, pp. 291-311, 2005.
4. E. Biham and N. Keller, *Cryptanalysis of Reduced Variants of Rijndael*, http://csrc.nist.gov/encryption/aes/round2/conf3/aes3papers.html.
5. E. Biham, O. Dunkelman and N. Keller, *The Rectangle Attack – Rectangling the Serpent*, Advances in Cryptology – Proceedings of EUROCRYPT 2001, LNCS 2045, pp. 340-357, Springer-Verlag, 2001.
6. E. Biham, O. Dunkelman and N. Keller, *Related-Key Boomerang and Rectangle Attacks*, Advances in Cryptology – Proceedings of EUROCRYPT 2005, LNCS 3494, pp. 507-525, Springer-Verlag, 2005.
7. E. Biham, O. Dunkelman and N. Keller, *A Related-Key Rectangle Attack on the Full KASUMI*, Advances in Cryptology – Proceedings of ASIACRYPT 2005, LNCS 3788, pp. 443-461, Springer-Verlag, 2005.
8. E. Biham, O. Dunkelman and N. Keller, *Related-Key Impossible Differential Attacks on AES-192*, Topics in Cryptology – Proceedings of CT-RSA 2006, LNCS 3860, pp. 21-31, Springer-Verlag, 2006.
9. A. Biryukov, *The Boomerang Attack on 5 and 6-Round AES*, Proceedings of AES 4, LNCS 3373, pp. 11-16, Springer-Verlag, 2005.
10. M. Blunden and A. Escott, *Related Key Attacks on Reduced Round KASUMI*, Proceedings of Fast Software Encryption 2001, LNCS 2355, pp. 277-285, Springer-Verlag, 2002.
11. J. Cheon, M. Kim, K. Kim, J. Lee and S. Kang, *Improved Impossible Differential Cryptanalysis of Rijndael and Crypton*, Proceedings of Information Security and Cryptology – ICISC 2001, LNCS 2288, pp. 39-49, Springer-Verlag, 2001.
12. J. Daemen and V. Rijmen, *The Design of Rijndael: AES – the Advanced Encryption Standard*, Springer-Verlag, 2002
13. O. Dunkelman, N. Keller and J. Kim, *Related-Key Rectangle Attack on the Full SHACAL-1*, Proceedings of Selected Areas in Cryptography 2006, to appear.
14. N. Ferguson, J. Kelsey, B. Schneier, M. Stay, D. Wagner and D. Whiting, *Improved Cryptanalysis of Rijndael*, Proceedings of Fast Software Encryption 2000, LNCS 1978, pp. 213-230, Springer-Verlag, 2001.

15. P. Hawkes, *Differential-Linear Weak-Key Classes of IDEA*, Advances in Cryptology – Proceedings of EUROCRYPT 1998, LNCS 1403, pp. 112-126, Springer-Verlag, 1998.
16. S. Hong, J. Kim, S. Lee and B. Preneel, *Related-Key Rectangle Attacks on Reduced Versions of SHACAL-1 and AES-192*, Proceedings of Fast Software Encryption 2005, LNCS 3557, pp. 368-383, Springer-Verlag, 2005.
17. G. Jakimoski and Y. Desmedt, *Related-Key Differential Cryptanalysis of 192-bit Key AES Variants*, Proceedings of Selected Areas in Cryptography 2003, LNCS 3006, pp. 208-221, Springer-Verlag, 2004.
18. J. Kelsey, B. Schneier and D. Wagner, *Key Schedule Cryptanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES*, Advances in Cryptology – Proceedings of CRYPTO 1996, LNCS 1109, pp. 237-251, Springer-Verlag, 1996.
19. J. Kelsey, B. Schneir and D. Wagner, *Related-Key Cryptanalysis of 3-WAY, Biham-DES, CAST, DES-X, NewDES, RC2, and TEA*, Proceedings of Information and Communications Seucrity – ICICS 1997, LNCS 1334, pp. 233-246, Springer-Verlag, 1997.
20. J. Kelsey, T. Kohno and B. Schneier, *Amplified Boomerang Attacks Against Reduced-Round MARS and Serpent*, Proceedings of Fast Software Encryption 2001, LNCS 1978, pp. 75-93, Springer-Verlag, 2002.
21. J. Kim, G. Kim, S. Hong, S. Lee and D. Hong, *The Related-Key Rectangle Attack – Application to SHACAL-1*, Proceedings of Information Security and Privacy – ACISP 2004, LNCS 3108, pp. 123-136, Springer-Verlag, 2004.
22. J. Kim, G. Kim, S. Lee, J. Lim and J. Song, *Related-Key Attacks on Reduced Rounds of SHACAL-2*, Proceedings of INDOCRYPT 2004, LNCS 3348, pp. 175-189, Springer-Verlag, 2004.
23. L.R. Knudsen, *Cryptanalysis of LOKI91*, Advances in Cryptology – Proceedings of AUSCRYPT 1992, LNCS 718, pp. 196-208, Springer-Verlag, 1993.
24. L.R. Knudsen, *Truncated and Higher Order Differentials*, Proceedings of Fast Software Encryption 1994, LNCS 1008, pp. 196-211, Springer-Verlag, 1995.
25. Y. Ko, S. Hong, W. Lee, S. Lee and J. Kang, *Related Key Differential Attacks on 26 Rounds of XTEA and Full Rounds of GOST*, Proceedings of Fast Software Encryption 2004, LNCS 3017, pp. 299-316, Springer-Verlag, 2004.
26. S.K. Langford and M.E. Hellman, *Differential-Linear Cryptanalysis*, Advances in Cryptology – Proceedings of CRYPTO 1994, LNCS 839, pp. 17-25, Springer-Verlag, 1994.
27. J. Lu, J. Kim, N. Keller and O. Dunkelman, *Related-Key Rectangle Attack on 42-Round SHACAL-2*, Proceedings of ISC 2006, LNCS 4176, pp. 85-100, Springer-Verlag, 2006.
28. J. Lu, C. Lee and J. Kim, *Related-Key Attacks on the Full-Round Cobra-F64a and Cobra-F64b*, Proceedings of SCN 2006, LNCS 4116, pp. 95-110, Springer-Verlag, 2006.
29. S. Lucks, *Attacking Seven Rounds of Rijndael under 192-bit and 256-bit Keys*, Proceedings of AES 3, NIST, 2000.
30. NESSIE — New European Schemes for Signatures, Integrity and Encryption, *Performance of Optimized Implementations of the NESSIE Primitives, version 2.0*, https://www.cosic.esat.kuleuven.be/nessie/deliverables/D21-v2.pdf
31. R. C.-W. Phan, *Impossible Differential Cryptanalysis of 7-round Advanced Encryption Standard (AES)*, Information Processing Letters, Volumn 91, Number 1, 33-38, Elsevier, 2004.
32. D. Wagner, *The Boomerang Attack*, Proceedings of Fast Software Encryption 1999, LNCS 1636, pp. 156-170, Springer-Verlag, 1999.