SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2019, Vol. 62 032101:1–032101:14
https://doi.org/10.1007/s11432-017-9365-4

Improved impossible differential cryptanalysis of large-block Rijndael

Ya LIU[1,2,3*], Yifan SHI[1], Dawu GU[3], Bo DAI[1], Fengyu ZHAO[1], Wei LI[4,5,6], Zhiqiang LIU[3,2] & Zhiqiang ZENG[7]

[1]*Engineering Research Center of Optical Instrument and System, Ministry of Education, Shanghai Key Lab of Modern Optical System, University of Shanghai for Science and Technology, Shanghai* 200093*, China;*
[2]*State Key Laboratory of Cryptology, P.O. Box* 5159*, Beijing* 100878*, China;*
[3]*Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai* 200240*, China;*
[4]*School of Computer Science and Technology, Donghua University, Shanghai* 201620*, China;*
[5]*Shanghai Key Laboratory of Scalable Computing and Systems, Shanghai* 200240*, China;*
[6]*Shanghai Key Laboratory of Integrated Administration Technologies for Information Security, Shanghai* 200240*, China;*
[7]*Science and Technology on Information Assurance Laboratory, Beijing* 100072*, China*

Received 28 November 2017/Accepted 31 January 2018/Published online 6 July 2018

Abstract Rijndael is a substitution-permutation network (SPN) block cipher for the AES development process. Its block and key sizes range from 128 to 256 bits in steps of 32 bits, which can be denoted by Rijndael-b-k, where b and k are the block and key sizes, respectively. Among them, Rijndael-128-128/192/256, that is, AES, has been studied by many researchers, and the security of other large-block versions of Rijndael has been exploited less frequently. However, more attention has been paid to large-block versions of block ciphers with the fast development of quantum computers. In this paper, we propose improved impossible differential attacks on 10-round Rijndael-256-256, 10-round Rijndael-224-256, and 9-round Rijndael-224-224 using precomputation tables, redundancies of key schedules, and multiple impossible differentials. For 10-round Rijndael-256-256, the data, time, and memory complexities of our attack were approximately $2^{244.4}$ chosen plaintexts, $2^{240.1}$ encryptions, and $2^{181.4}$ blocks, respectively. For 10-round Rijndael-224-256, the data, time, and memory complexities of our attack were approximately $2^{214.4}$ chosen plaintexts, $2^{241.3}$ encryptions, and $2^{183.4}$ blocks, respectively. For 9-round Rijndael-224-224, the data, time, and memory complexities of our attack are approximately $2^{214.4}$ chosen plaintexts, $2^{113.4}$ encryptions, and $2^{87.4}$ blocks, respectively, or $2^{206.6}$ chosen plaintexts, $2^{153.6}$ encryptions, and $2^{111.6}$ blocks, respectively. To the best of our knowledge, our results are currently the best on Rijndael-256-256 and Rijndael-224-224/256.

Keywords block cipher, Rijndael, precomputation tables, impossible differentials, multiple impossible differential attacks

Citation Liu Y, Shi Y F, Gu D W, et al. Improved impossible differential cryptanalysis of large-block Rijndael. Sci China Inf Sci, 2019, 62(3): 032101, https://doi.org/10.1007/s11432-017-9365-4

1 Introduction

In 1997, Daemen and Rijmen [1] designed a substitution-permutation network (SPN) block cipher, Rijndael, for the Advanced Encryption Standard (AES) development process. It has variable block and key sizes that range from 128 to 256 bits in steps of 32 bits. It is well known that the 128-bit block version of Rijndael with key sizes of 128, 192, and 256 bits was selected as the AES by National Institute

* Corresponding author (email: liuya@usst.edu.cn)

of Standards and Technology (NIST) in 2002. The other large-block versions of Rijndael are denoted by Rijndael-b-k with block size $b \in \{160, 192, 224, 256\}$ and key size $k \in \{128, 160, 192, 224, 256\}$. In all versions, the number of rounds depends on the block and key sizes, which vary from 10 to 14. To date, there have been many attacks that estimate the security of the AES, such as square attacks, impossible differential attacks, boomerang attacks, rectangle attacks, and meet-in-the-middle attacks [2–13]. With respect to the AES, there have been fewer attacks used to exploit the security of large-block versions of Rijndael, including multiset, integral attacks, and impossible differential attacks [14–16]. However, the fast development of quantum computers has greatly challenged the security of block ciphers with small key sizes because Grover's quantum algorithm [17] can provide a quadratic speedup for the exhaustive search. Considering a high level of safety, more attention has been paid to large-block versions of block ciphers in the industrial community. Therefore, it is very important to study the security of large-block versions of Rijndael.

Impossible differential cryptanalysis was independently proposed by Knudsen [18] and Biham in 1999 [14]. Its fundamental concept is to construct a differential path with probability zero (called impossible differentials) to eliminate all wrong candidate keys until the correct key is retrieved. Specifically, the adversaries first construct two truncated differentials with probability one from the plaintext direction and ciphertext direction. These two differentials contradict each other in the middle and then are combined into an impossible differential. By adding rounds before and/or after this distinguisher, the attackers construct an attacking path. Next, plaintext-ciphertexts are selected and the round subkeys are guessed. If there is a plaintext-ciphertext that satisfies the input and output differences of the impossible differential under some guessed round subkey, this round subkey is wrong. Given a sufficient number of plaintext-ciphertexts, all wrong subkeys can be removed from the key space and the right subkey can be retrieved. To date, many new results have been presented to improve its efficiency, such as the early abort technique [19], state-test technique [20], pre-computation tables [21], and automated algorithms for impossible differentials [22–26]. As one of the most powerful attacks, impossible cryptanalysis can be used to estimate the security of large-block Rijndael and obtain its previous best results. Specifically, in 2007, Nakahara and Pavao [27] proposed impossible differential attacks on seven rounds of Rijndael-224-224 and seven rounds of Rijndael-256-256. In 2008, Zhang et al. [28] presented the impossible differential cryptanalysis of 9-round Rijndael-224-224 and 9-round Rijndael-256-256. In 2012, Wang et al. [29] improved previous results and mounted impossible differential attacks on 9-round Rijndael-224-224 and 10-round Rijndael-256-256. Recently, Minier [30] proposed two new impossible differential attacks on 8-round Rijndael-160-192 and 10-round Rijndael-224-256. However, he used the general formulas in [20, 31] to estimate the complexities of the attack. Unfortunately, researchers [32] have indicated that the complexities that are calculated by applying these formulas are smaller than the real values.

In this paper, we improve previous results and propose multiple impossible differential attacks on 10-round Rijndael-256-256, 10-round Rijndael-224-256, and 9-round Rijndael-224-224. In the key recovery phase, we build precomputation tables to extract the related round keys involved in the analysis rounds to reduce the time complexity. For Rijndael-256-256, we apply eight 6-round impossible differentials to attack 10-round Rijndael-256-256 with $2^{244.4}$ chosen plaintexts, $2^{240.1}$ encryptions, and $2^{181.4}$ blocks. For Rijndael-224-256, we use four 6-round impossible differentials to attack 10-round Rijndael-224-256 with $2^{214.4}$ chosen plaintexts, $2^{241.3}$ encryptions, and $2^{183.4}$ blocks, and 9-round Rijndael-224-224 with $2^{206.6}$ chosen plaintexts, $2^{153.6}$ encryptions, and $2^{111.6}$ blocks or $2^{214.4}$ chosen plaintexts, $2^{113.4}$ encryptions, and $2^{87.4}$ blocks. Compared with the previously best attacks [29], the time and memory complexities of our attack on 10-round Rijndael-256-256 are reduced by $2^{13.8}$ times and $2^{5.4}$ times, respectively, the data, time and memory complexities of our attacks on 9-round Rijndael-224-224 are reduced by $2^{1.4}$ times, $2^{8.4}$ times and $2^{5.4}$ times, respectively, or $2^{1.6}$ times, $2^{16.6}$ times and $2^{6.2}$ times, respectively. Meanwhile, we propose an attack on 10-round Rijndael-224-256 for the first time. In Table 1, our results and previous results for Rijndael-224-224/256 and Rijndael-256-256 are listed, except for paper [30]. NR, IDA, IA, CP, and Enc denote the number of rounds, impossible differential attacks, integral attacks, chosen plaintexts, and encryptions, respectively. To the best of our knowledge, these results are the best attacks on Rijndael-224-224/256 and Rijndael-256-256.

Table 1 Summary of attacks on Rijndael-256-256 and Rijndael-224-256

Cipher	NR	Data (CP)	Time (Enc)	Memory (Blocks)	Attack type	Source
Rijndael-256-256	7	2^{153}	2^{182}	2^{117}	IDA	[27]
	9	$2^{244.3}$	$2^{208.8}$	2^{192}	IDA	[28]
	9	$2^{132.5}$	$2^{174.5}$	–	IA	[33]
	9	$2^{237.3}$	$2^{159.1}$	$2^{115.3}$	IDA	[29]
	9	$2^{245.3}$	$2^{127.1}$	$2^{90.9}$	IDA	[29]
	10	$2^{244.2}$	$2^{253.9}$	$2^{186.8}$	IDA	[29]
	10	$2^{244.4}$	$2^{240.1}$	$2^{181.4}$	IDA	Subsection 3.2
Rijndael-224-256	10	$2^{214.4}$	$2^{241.3}$	$2^{183.4}$	IDA	Subsection 4.2
Rijndael-224-224	7	2^{138}	2^{167}	2^{104}	IDA	[27]
	9	$2^{212.3}$	2^{209}	2^{192}	IDA	[28]
	9	$2^{196.5}$	$2^{196.5}$	–	IA	[33]
	9	2^{208}	2^{162}	2^{117}	IDA	[29]
	9	2^{216}	2^{130}	$2^{93.6}$	IDA	[29]
	9	$2^{214.4}$	$2^{113.4}$	$2^{87.4}$	IDA	Subsection 4.3
	9	$2^{206.6}$	$2^{153.6}$	$2^{111.6}$	IDA	Subsection 4.3

The remainder of this paper is organized as follows. In Section 2, we introduce the preliminaries. In Section 3, we propose an impossible differential attack on 10-round Rijndael-256-256. In Section 4, we propose impossible differential attacks on 10-round Rijndael-224-256 and 9-round Rijndael-224-224. In Section 5, we summarize this paper.

2 Description of Rijndael

2.1 Notations

- P, C, K: plaintext, ciphertext, and master key.
- $\Delta P, \Delta C$: plaintext and ciphertext differences.
- X_i, Y_i, Z_i, W_i: intermediate states after the MixColumn (MC), SubByte (SB), ShiftRow (SR), AddRoundKey (ARK) operations in the i-th round, respectively.
- $X_i[j], Y_i[j], Z_i[j], W_i[j]$: the j-th byte of X_i, Y_i, Z_i, W_i, where $0 \leqslant j < 16$.
- $\Delta X_i, \Delta Y_i, \Delta Z_i, \Delta W_i$: differences of X_i, Y_i, Z_i, W_i, respectively.
- $0_{(n)}$: n bits of zero in parallel.
- ARK_i: subkey bits in the i-th round.
- ARK_i^*: equivalent round subkeys when the order of MixColumn and AddRoundKey is exchanged, that is, $\mathrm{ARK}_i^* = \mathrm{MC}^{-1}(\mathrm{ARK}_i)$.
- $\mathrm{ARK}_i[j], \mathrm{ARK}_i^*[j]$: j-th byte of ARK_i and ARK_i^* with $0 \leqslant j < 16$.
- W_i^*: intermediate states after the ARK^* operations when the order of MixColumn and AddRoundKey is exchanged in the i-th round.
- $W_i^*[j]$: j-th byte of W_i^*, where $0 \leqslant j < 16$.
- ΔW_i^*: difference of W_i^*.
- $A \parallel B$: concatenation of A and B.

2.2 Rijndael

Rijndael is an SPN block cipher. It supports variable block and key sizes, which can range from 128 to 256 bits in steps of 32 bits. The number of rounds is 10, 12, or 14 depending on the text and key lengths. All versions can be denoted as Rijndael-b-k, where b and k are the block size and key size, respectively. Additionally, the states include plaintext, ciphertext, and round subkeys, and all intermediate states can be described as a $4 \times N_b$ matrix with four rows and N_b $(= b/32)$ columns. The master key is similarly represented as a matrix with four rows and N_k $(= k/32)$ columns. The round function consists

of SubByte (SB), ShiftRow (SR), MixColumn (MC), and AddRoundKey (ARK). Before the first round, an extra ARK is added. In the final round, MC is discarded.

- SB: Nonlinear transformation applies 8×8 S-boxes in parallel.
- SR: Each row shifts to the left over the different offsets, and shift offset C_i of row i depends on N_b. For Rijndael-224-224/256, $(C_0, C_1, C_2, C_3) = (0, 1, 2, 4)$. For Rijndael-256-256, $(C_0, C_1, C_2, C_3) = (0, 1, 3, 4)$.
- MC: Each column of internal data is updated by matrix M, whose branch number is five.
- ARK: All the round keys are XORed to the internal dates.

Key schedule. If Rijndael has N_r rounds, it requires $N_r + 1$ round subkeys that are made from the master key, that is, the master key is assigned to the first N_k words $W[0] \parallel W[1] \parallel \cdots \parallel W[N_k - 1]$ directly, whereas the remaining round subkey words $W[i]$ for $i \in \{N_k, \ldots, N_k \times (N_r+1)-1\}$ are generated by Algorithm 1. Among them, f and g are nonlinear permutations and $\mathrm{rcon}[i/N_k]$ denotes fixed round constants. We have only described Rijndael briefly. More details can be found in [1, 2].

Algorithm 1 Key schedule

if $i \mod N_k = 0$ **then**
  $W[i] = W[i - N_k] \oplus f(W[i-1]) \oplus \mathrm{rcon}[i/Nk]$
else
  if $((N_k > 6)$ and $(i \mod N_k = 4))$ **then**
    $W[i] = W[i - N_k] \oplus g(W[i-1])$
  else
    $W[i] = W[i - N_k] \oplus W[i-1]$
  end if
end if

3 Improved impossible differential attacks on Rijndael-256-256

3.1 6-Round impossible differentials of Rijndael-256-256

In this section, we present 64 6-round impossible differentials in Proposition 1. In our attacks, we exchange the MC and ARK operations to reduce the number of guessed round keys. The new encryption algorithm is equivalent to the previous algorithm.

Proposition 1 ([29]). In Rijndael-256-256, there are 64 impossible differentials. The input difference of the impossible differentials has 32 scenarios, that is, one non-zero (active) byte can be set in any cell, and the other bytes are inactive. The output difference of the impossible differentials has two options: three active bytes at $(0, 1, 3)$ or $(0, 2, 3)$ and the remaining bytes are inactive. In our attack on 10-round Rijndael-256-256, we apply eight impossible differentials as follows:

$$\Delta_1 : (\alpha, 0_{(8)}, 0_{(8)}, 0_{(8)}, 0_{(224)}) \nrightarrow (\beta, \gamma, 0_{(8)}, \delta, 0_{(224)}),\ \Delta_2 : (\alpha, 0_{(8)}, 0_{(8)}, 0_{(8)}, 0_{(224)}) \nrightarrow (\beta, 0_{(8)}, \gamma, \delta, 0_{(224)}),$$

$$\Delta_3 : (0_{(8)}, \alpha, 0_{(8)}, 0_{(8)}, 0_{(224)}) \nrightarrow (\beta, \gamma, 0_{(8)}, \delta, 0_{(224)}),\ \Delta_4 : (0_{(8)}, \alpha, 0_{(8)}, 0_{(8)}, 0_{(224)}) \nrightarrow (\beta, 0_{(8)}, \gamma, \delta, 0_{(224)}),$$

$$\Delta_5 : (0_{(8)}, 0_{(8)}, \alpha, 0_{(8)}, 0_{(224)}) \nrightarrow (\beta, \gamma, 0_{(8)}, \delta, 0_{(224)}),\ \Delta_6 : (0_{(8)}, 0_{(8)}, \alpha, 0_{(8)}, 0_{(224)}) \nrightarrow (\beta, 0_{(8)}, \gamma, \delta, 0_{(224)}),$$

$$\Delta_7 : (0_{(8)}, 0_{(8)}, 0_{(8)}, \alpha, 0_{(224)}) \nrightarrow (\beta, \gamma, 0_{(8)}, \delta, 0_{(224)}),\ \Delta_8 : (0_{(8)}, 0_{(8)}, 0_{(8)}, \alpha, 0_{(224)}) \nrightarrow (\beta, 0_{(8)}, \gamma, \delta, 0_{(224)}),$$

where α, β, γ, and δ are non-zero bytes. We show Δ_1 in Figure 1.

3.2 Impossible differential attacks on 10-round Rijndael-256-256

Proposition 2 ([34]). Given the input and output differences $\Delta_i \in F_{256}^*$ and $\Delta_o \in F_{256}^*$, respectively, the equation $S(x) \oplus S(x \oplus \Delta_i) = \Delta_o$ has one solution, on average.

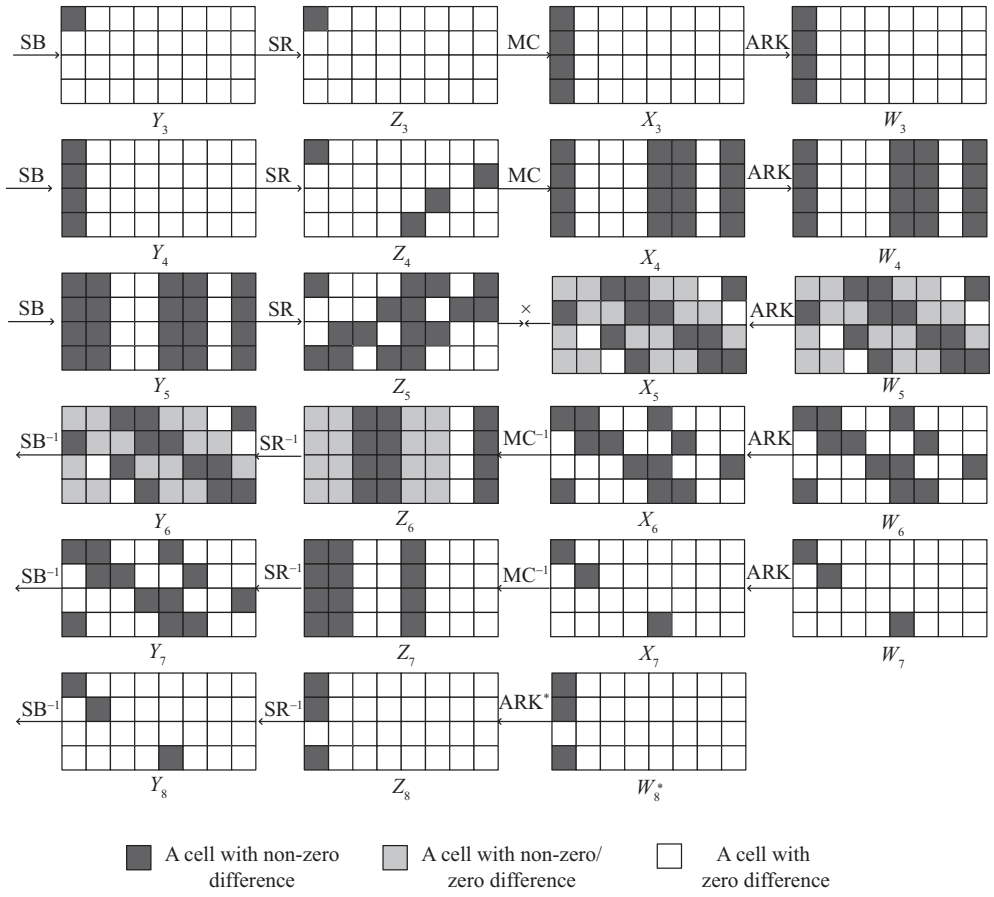

Figure 1 A 6-round impossible differential path of Rijndael-256-256.

Proposition 3 ([35]). If the values (or differences) in any four out of the eight input/output bytes of the MixColumn operation are known, then the values (or differences) in the other four bytes are uniquely determined and can be computed efficiently.

Based on 6-round impossible differentials, we add two rounds both at the top and bottom to attack 10-round Rijndael-256-256, as shown in Figure 2. The attack procedure contains two phases: data collection and key recovery. In the data collection phase, we select plaintext-ciphertext pairs that satisfy the input and output differences. In the key recovery phase, we build precomputation tables to remove the wrong subkeys. The detailed attack procedure is provided in the following.

Data collection. We choose 2^n structures of plaintexts, each of which takes all the possible values at $(0, 3, 4, 5, 9, 12, 14, 16, 17, 18, 19, 21, 23, 26, 30, 31)$ bytes, and the other bytes are fixed. Thus, a structure contains 2^{128} plaintexts that can form approximately 2^{255} pairs of plaintexts. We encrypt these plaintexts and build a hash table. In this table, we store those plaintext-ciphertext pairs whose ciphertext differences are non-zero at $(0, 15, 18, 19, 22, 25, 28, 29)$ bytes. Finally, $2^{-8\times24} \times 2^{n+255} = 2^{n+63}$ pairs remain.

Key recovery. Before presenting key recovery, we build precomputation tables to extract the related round subkeys so that the time complexity is reduced.

- H_1: Guess the values of $\Delta W_9^*[0] \parallel W_9^*[0, 1, 2, 3]$. Compute the values of $Z_{10}[0, 29, 22, 19] \parallel \Delta Z_{10}[0, 29, 22, 19]$. Store $Z_{10}[0, 29, 22, 19] \parallel W_9^*[0] \parallel \Delta W_9^*[0]$ in H_1 indexed by $\Delta C[0, 29, 22, 19]$, where $\Delta C[0, 29, 22, 19] = \Delta Z_{10}[0, 29, 22, 19]$. Thus, H_1 has 2^{32} rows, each of which has 2^8 values, on average.
- H_2: Guess the values of $\Delta W_9^*[29] \parallel W_9^*[28, 29, 30, 31]$. Compute the values of $Z_{10}[28, 25, 18, 15] \parallel \Delta Z_{10}[28, 25, 18, 15]$. Store $Z_{10}[28, 25, 18, 15] \parallel \Delta W_9^*[29] \parallel W_9^*[29]$ in H_2 indexed by $\Delta C[28, 25, 18, 15]$, where $\Delta C[28, 25, 18, 15] = \Delta Z_{10}[28, 25, 18, 15]$. Thus, H_2 has 2^{32} rows, each of which has 2^8 values, on average.
- H_3^a (for the active bytes of the output difference of the impossible differentials at (0,1,3)): Guess the

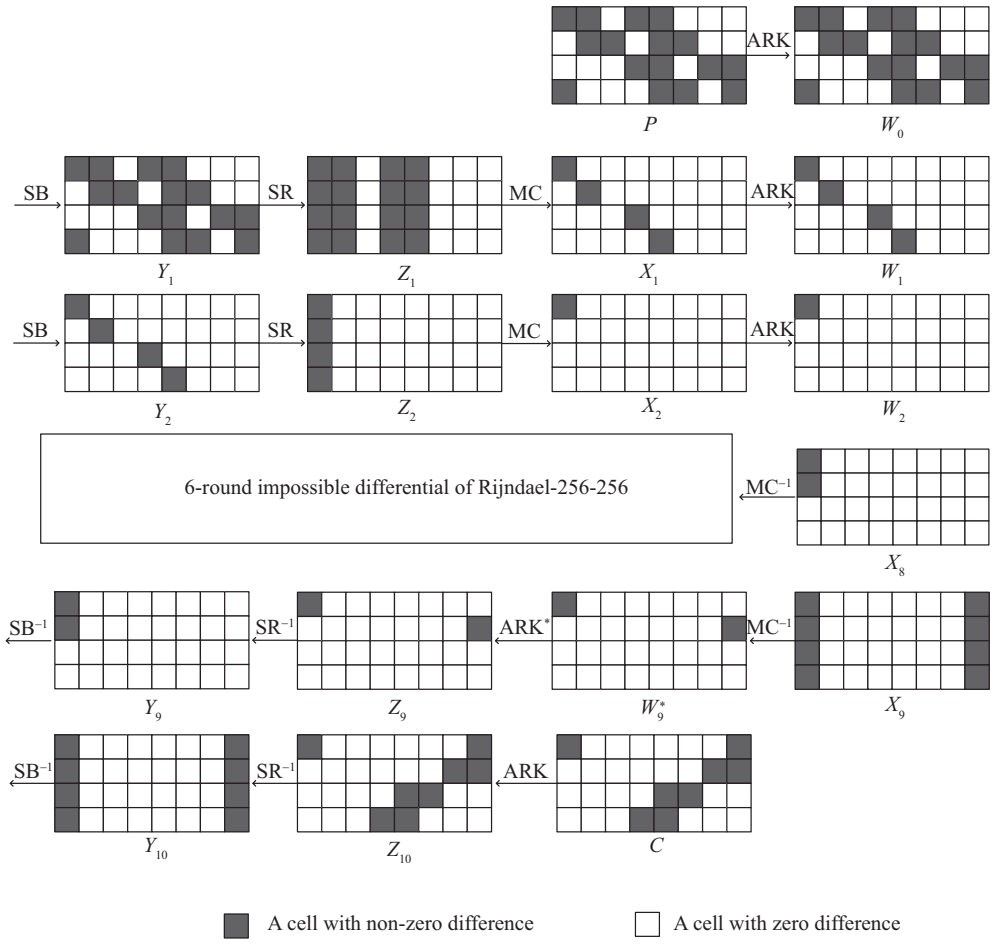

Figure 2 Impossible differential attacks on 10-round Rijndael-256-256.

values of $\Delta X_8[0]$. Because $\Delta X_8[i] = 0 (i = 2, 3)$ and $\Delta W_8^*[2] = 0$, the value of $\Delta X_8[1]$ can be computed. Next, guess 2^{16} values of $X_8[0, 1]$ to deduce the values of $Z_9[0, 29] \parallel \Delta Z_9[0, 29]$. Store the values of $Z_9[0, 29]$ in H_3^a indexed by $\Delta W_9^*[0, 29]$. Thus, H_3^a has 2^{16} rows, each of which has 2^8 values, on average.

• H_3^b (for the active bytes of the output difference of the impossible differentials at (0,2,3)): Guess the values of $\Delta X_8[0]$. Because $\Delta X_8[i] = 0 (i = 2, 3)$ and $\Delta W_8^*[1] = 0$, the value of $\Delta X_8[1]$ can be computed. Next, guess 2^{16} values of $X_8[0, 1]$ to deduce the values of $Z_9[0, 29] \parallel \Delta Z_9[0, 29]$. Store the values of $Z_9[0, 29]$ in H_3^b indexed by $\Delta W_9^*[0, 29]$. Thus, H_3^b has 2^{16} rows, each of which has 2^8 values, on average.

• H_4: Guess the values of $\Delta X_1[0] \parallel X_1[0, 1, 2, 3]$. Compute the values of $W_0[0, 5, 14, 19] \parallel \Delta W_0[0, 5, 14, 19]$. Store $W_0[0, 5, 14, 19] \parallel \Delta X_1[0] \parallel X_1[0]$ in H_4 indexed by $\Delta P[0, 5, 14, 19]$. Thus, H_4 has 2^{32} rows, each of which has 2^8 values, on average.

• H_5: Guess the values of $\Delta X_1[5] \parallel X_1[4, 5, 6, 7]$, and compute the values of $W_0[4, 9, 18, 23] \parallel \Delta W_0[4, 9, 18, 23]$. Store $W_0[4, 9, 18, 23] \parallel \Delta X_1[5] \parallel X_1[5]$ in H_5 indexed by $\Delta P[4, 9, 18, 23]$. Thus, H_5 has 2^{32} rows, each of which has 2^8 values, on average.

• H_6: Guess the values of $\Delta X_1[14] \parallel X_1[12, 13, 14, 15]$. Compute the values of $W_0[12, 17, 26, 31] \parallel \Delta W_0[12, 17, 26, 31]$. Store $W_0[12, 17, 26, 31] \parallel \Delta X_1[14] \parallel X_1[14]$ in H_6 indexed by $\Delta P[12, 17, 26, 31]$. Thus, H_6 has 2^{32} rows, each of which has 2^8 values, on average.

• H_7: Guess the values of $\Delta X_1[19] \parallel X_1[16, 17, 18, 19]$. Compute the values of $W_0[16, 21, 30, 3] \parallel \Delta W_0[16, 21, 30, 3]$. Store $W_0[16, 21, 30, 3] \parallel \Delta X_1[19] \parallel X_1[19]$ in H_7 indexed by $\Delta P[16, 21, 30, 3]$. Thus, H_7 has 2^{32} rows, each of which has 2^8 values, on average.

• H_8^a (for the active bytes of the input difference of the impossible differentials at 0): Guess the values of $\Delta X_2[0] \parallel X_2[0, 1, 2, 3]$. Compute the values of $W_1[0, 5, 14, 19] \parallel \Delta W_1[0, 5, 14, 19]$. Store $W_1[0, 5, 14, 19]$ in H_8^a indexed by $\Delta W_1[0, 5, 14, 19]$. Thus, H_8^a has 2^{32} rows, each of which has 2^8 values, on average.

• H_8^b (for the active bytes of the input difference of the impossible differentials at 1): Guess the values of $\Delta X_2[1] \parallel X_2[0,1,2,3]$. Compute the values of $W_1[0,5,14,19] \parallel \Delta W_1[0,5,14,19]$. Store $W_1[0,5,14,19]$ in H_8^b indexed by $\Delta W_1[0,5,14,19]$. Thus, H_8^b has 2^{32} rows, each of which has 2^8 values, on average.

• H_8^c (for the active bytes of the input difference of the impossible differentials at 2): Guess the values of $\Delta X_2[2] \parallel X_2[0,1,2,3]$. Compute the values of $W_1[0,5,14,19] \parallel \Delta W_1[0,5,14,19]$. Store $W_1[0,5,14,19]$ in H_8^c indexed by $\Delta W_1[0,5,14,19]$. Thus, H_8^c has 2^{32} rows, each of which has 2^8 values, on average.

• H_8^d (for the active bytes of input difference of impossible differentials at 3): Guess the values of $\Delta X_2[3] \parallel X_2[0,1,2,3]$. Compute the values of $W_1[0,5,14,19] \parallel \Delta W_1[0,5,14,19]$. Store $W_1[0,5,14,19]$ in H_8^d indexed by $\Delta W_1[0,5,14,19]$. Thus, H_8^d has 2^{32} rows, each of which has 2^8 values, on average.

The round subkeys can be retrieved as follows.

(1) Access H_1 to compute $\mathrm{ARK}_{10}[0,29,22,19]$. Because $\mathrm{ARK}_{10}[0,29,22,19] = Z_{10}[0,29,22,19] \oplus C[0,29,22,19]$. Thus, we obtain 2^8 values of $\mathrm{ARK}_{10}[0,29,22,19]$.

(2) Access H_2 to compute $\mathrm{ARK}_{10}[28,25,18,15]$. Because $\mathrm{ARK}_{10}[28,25,18,15] = Z_{10}[28,25,18,15] \oplus C[28,25,18,15]$. Thus, there are $2^{8+8} = 2^{16}$ values of $\mathrm{ARK}_{10}[0,29,22,19] \parallel \mathrm{ARK}_{10}[28,25,18,15]$.

(3) Access H_3^a and H_3^b to compute $\mathrm{ARK}_9^*[0,29]$. Because $\mathrm{ARK}_9^*[0,29] = Z_9[0,29] \oplus W_9^*[0,29]$, there are $2^8 + 2^8 = 2^9$ possible values for $\mathrm{ARK}_9^*[0,29]$. Thus, there are $2^{16+9} = 2^{25}$ values of $\mathrm{ARK}_{10}[0,29,22,19] \parallel \mathrm{ARK}_{10}[28,25,18,15] \parallel \mathrm{ARK}_9^*[0,29]$.

(4) Access H_4 to compute $\mathrm{ARK}_0[0,5,14,19]$. Because $\mathrm{ARK}_0[0,5,14,19] = X_0[0,5,14,19] \oplus P[0,5,14,19]$, we obtain $2^{25+8} = 2^{33}$ values of $\mathrm{ARK}_{10}[0,29,22,19] \parallel \mathrm{ARK}_{10}[28,25,18,15] \parallel \mathrm{ARK}_9^*[0,29] \parallel \mathrm{ARK}_0[0,5,14,19]$.

(5) Access H_5 to compute $\mathrm{ARK}_0[4,9,18,23]$. Because $\mathrm{ARK}_0[4,9,18,23] = X_0[4,9,18,23] \oplus P[4,9,18,23]$, we obtain $2^{33+8} = 2^{41}$ values of $\mathrm{ARK}_{10}[0,29,22,19] \parallel \mathrm{ARK}_{10}[28,25,18,15] \parallel \mathrm{ARK}_9^*[0,29] \parallel \mathrm{ARK}_0[0,5,14,19] \parallel \mathrm{ARK}_0[4,9,18,23]$.

(6) Access H_6 to compute $\mathrm{ARK}_0[12,17,26,31]$. Because $\mathrm{ARK}_0[12,17,26,31] = X_0[12,17,26,31] \oplus P[12,17,26,31]$, we obtain $2^{41+8} = 2^{49}$ values of $\mathrm{ARK}_{10}[0,29,22,19] \parallel \mathrm{ARK}_{10}[28,25,18,15] \parallel \mathrm{ARK}_9^*[0,29] \parallel \mathrm{ARK}_0[0,5,14,19] \parallel \mathrm{ARK}_0[4,9,18,23] \parallel \mathrm{ARK}_0[12,17,26,31]$.

(7) Access H_7 to compute $\mathrm{ARK}_0[16,21,30,3]$. Because $\mathrm{ARK}_0[16,21,30,3] = X_0[16,21,30,3] \oplus P[16,21,30,3]$, we get $2^{49+8} = 2^{57}$ values of $\mathrm{ARK}_{10}[0,29,22,19] \parallel \mathrm{ARK}_{10}[28,25,18,15] \parallel \mathrm{ARK}_9^*[0,29] \parallel \mathrm{ARK}_0[0,5,14,19] \parallel \mathrm{ARK}_0[4,9,18,23] \parallel \mathrm{ARK}_0[12,17,26,31] \parallel \mathrm{ARK}_0[16,21,30,3]$.

(8) Access H_8^a, H_8^b, H_8^c and H_8^d to compute $\mathrm{ARK}_1[0,5,14,19]$. Because $\mathrm{ARK}_1[0,5,14,19] = W_1[0,5,14,19] \oplus X_1[0,5,14,19]$, there are 2^{10} possible values of $\mathrm{ARK}_1[0,5,14,19]$. Finally, we obtain $2^{57+10} = 2^{67}$ values of $\mathrm{ARK}_{10}[0,29,22,19] \parallel \mathrm{ARK}_{10}[28,25,18,15] \parallel \mathrm{ARK}_9^*[0,29] \parallel \mathrm{ARK}_0[0,5,14,19] \parallel \mathrm{ARK}_0[4,9,18,23] \parallel \mathrm{ARK}_0[12,17,26,31] \parallel \mathrm{ARK}_0[16,21,30,3] \parallel \mathrm{ARK}_1[0,5,14,19]$.

There are 30 bytes of round subkeys involved in our attack. For each plaintext-ciphertext pair, approximately 2^{67} possible round subkeys are removed. Thus, the probability that a wrong key is not discarded for one plaintext-ciphertext pair is $1 - 2^{67-240} = 1 - 2^{-173}$. After filtering through 2^{n+63} pairs, there are approximately $2^{240} \times (1 - 2^{-173})^{2^{n+63}} = 2^{120}$ remaining candidates for 128 bits of the master key when $n = 116.4$. Because of the key schedule, $\mathrm{ARK}_0[1,29]$ is calculated by $\mathrm{ARK}_0[29] = \mathrm{ARK}_0[0] \oplus \mathrm{ARK}_1[0]$ and $\mathrm{ARK}_0[1] = \mathrm{ARK}_0[30] \oplus \mathrm{ARK}_1[1]$, $\mathrm{ARK}_1[1]$ can be deduced by $\mathrm{ARK}_1[1] = \mathrm{ARK}_0[5] \oplus \mathrm{ARK}_1[5]$. The remaining 14 bytes of the master keys are guessed exhaustively. The time complexity of this step is approximately $2 \times 2^{120} \times 2^{112} \approx 2^{233}$. The time complexity to access the precomputation tables is dominated by the last step, that is, approximately $2^{116.4+63+67} \div (8 \times 10) \approx 2^{240.1}$. Therefore, the entire time complexity is approximately $2^{240.1}$ encryptions, the data complexity is approximately $2^{116.4+128} = 2^{244.4}$ chosen plaintexts, and the memory complexity is approximately $2^{116.4+63} \times 4 = 2^{181.4}$ 256-bit blocks.

4 Improved impossible differential attacks on Rijndael-224-224/256

In this section, we attack 10-round Rijndael-224-256 and 9-round Rijndael-224-224 based on 6-round impossible differential paths.

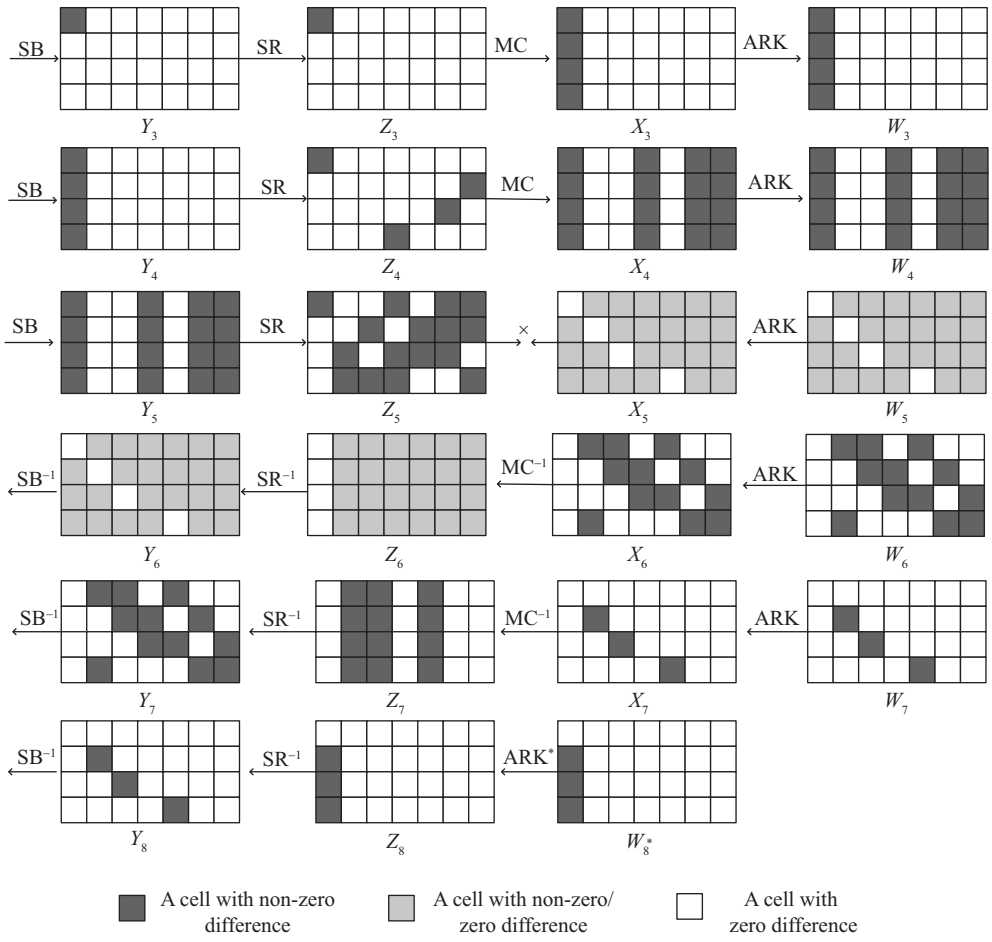

Figure 3 A 6-round impossible differential path of Rijndael-224-224/256.

4.1 6-Round impossible differentials of Rijndael-224-224/256

Similarly, we construct four 6-round impossible differentials of Rijndael-224-224/256 in Proposition 4.

Proposition 4 ([29])**.** In our attacks on Rijndael-224-224/256, we use four impossible differentials as follows:

$$\Delta_1' : (\alpha, 0_{(8)}, 0_{(8)}, 0_{(8)}, 0_{(192)}) \nrightarrow (0_{(8)}, \beta, \gamma, \delta, 0_{(192)}), \Delta_2' : (0_{(8)}, \alpha, 0_{(8)}, 0_{(8)}, 0_{(192)}) \nrightarrow (0_{(8)}, \beta, \gamma, \delta, 0_{(192)}),$$

$$\Delta_3' : (0_{(8)}, 0_{(8)}, \alpha, 0_{(8)}, 0_{(192)}) \nrightarrow (0_{(8)}, \beta, \gamma, \delta, 0_{(192)}), \Delta_4' : (0_{(8)}, 0_{(8)}, 0_{(8)}, \alpha, 0_{(192)}) \nrightarrow (0_{(8)}, \beta, \gamma, \delta, 0_{(192)}),$$

where α, β, γ, and δ are non-zero bytes. In Figure 3, we show the structure of Δ_1'.

4.2 Impossible differential attacks on 10-round Rijndael-224-256

Based on the 6-round impossible differentials of Rijndael-224-224/256 in Subsection 4.1, we add two rounds at the top and two rounds at the bottom to attack 10-round Rijndael-224-256, as shown in Figure 4. The attack procedure is provided in the following.

Data collection. We choose 2^n structures of plaintexts, each of which takes all the possible values at $(0, 4, 5, 7, 8, 9, 10, 13, 14, 16, 18, 19, 21, 23, 26, 27)$ bytes and the other bytes are fixed. Thus, a structure contains 2^{128} plaintexts that form approximately 2^{255} pairs of plaintexts. We encrypt these plaintexts and build a hash table to store the plaintext-ciphertext pairs whose ciphertext differences are non-zero at $(0, 11, 15, 18, 21, 22, 24, 25)$ bytes. Finally, $2^{-8\times 20} \times 2^{n+255} = 2^{n+95}$ pairs remain.

Key recovery. Before describing key recovery, we build precomputation tables to extract the related round subkeys.

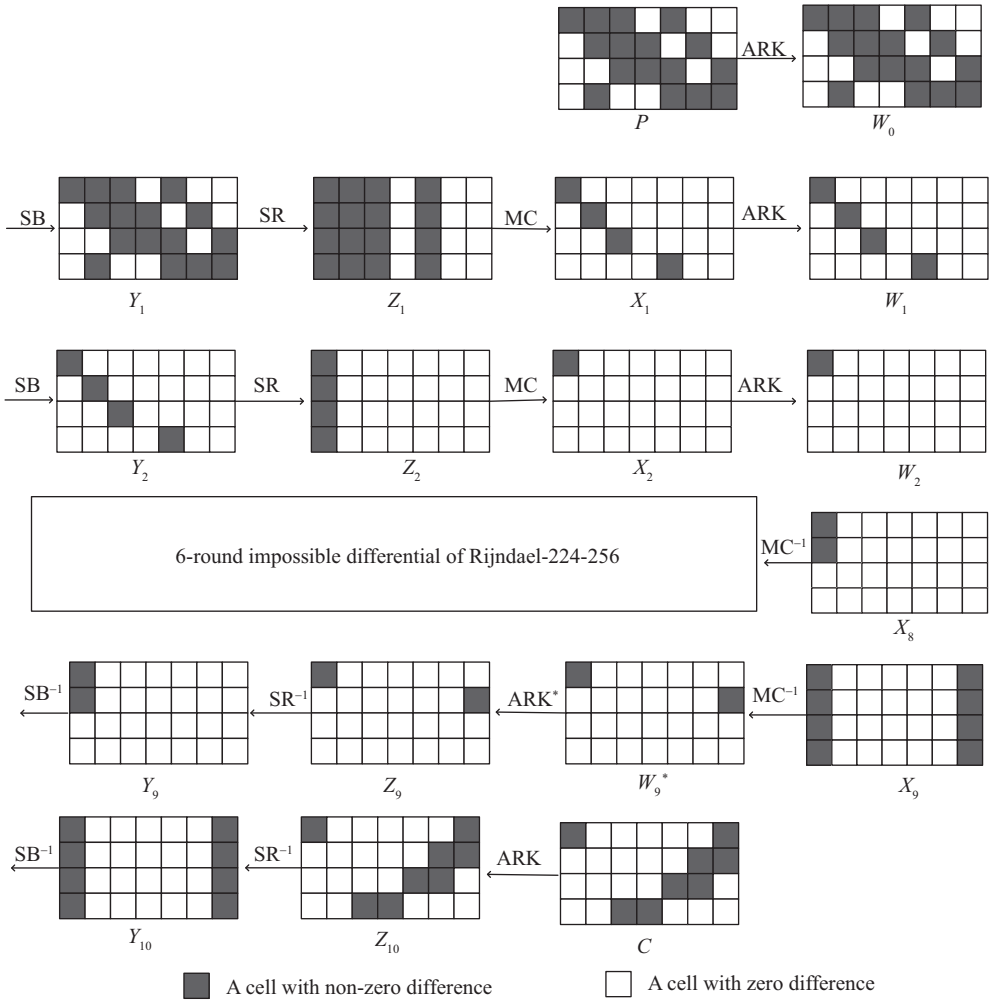

Figure 4 Impossible differential attacks on 10-round Rijndael-224-256.

• H_1: Guess the values of $\Delta W_9^*[0] \parallel W_9^*[0,1,2,3]$. Compute the value of $Z_{10}[0,15,22,25] \parallel \Delta Z_{10}[0,15,22,25]$. Store $Z_{10}[0,15,22,25] \parallel W_9^*[0] \parallel \Delta W_9^*[0]$ in H_1 indexed by $\Delta C[0,15,22,25]$, where $\Delta C[0,15,22,25] = \Delta Z_{10}[0,15,22,25]$. Thus, H_1 has 2^{32} rows, each of which has 2^8 values, on average.

• H_2: Guess the values of $\Delta W_9^*[25] \parallel W_9^*[24,25,26,27]$. Compute the value of $Z_{10}[11,18,21,24] \parallel \Delta Z_{10}[11,18,21,24]$. Store $Z_{10}[11,18,21,24] \parallel \Delta W_9^*[25] \parallel W_9^*[25]$ in H_2 indexed by $\Delta C[11,18,21,24]$, where $\Delta C[11,18,21,24] = \Delta Z_{10}[11,18,21,24]$. Thus, H_2 has 2^{32} rows, each of which has 2^8 values, on average.

• H_3 : Guess the values of $\Delta X_8[0]$. Because $\Delta X_8[i] = 0 (i = 2,3)$ and $\Delta W_8^*[0] = 0$, the value of $\Delta X_8[1]$ can be computed. Next, guess 2^{16} values of $X_8[0,1]$ to deduce the values of $Z_9[0,25] \parallel \Delta Z_9[0,25]$. Store the values of $Z_9[0,25]$ in H_3^a indexed by $\Delta W_9[0,25]$. Thus, H_3^a has 2^{16} rows, each of which has 2^8 values, on average.

• H_4: Guess the values of $\Delta X_1[0] \parallel X_1[0,1,2,3]$. Compute the value of $W_0[0,5,10,19] \parallel \Delta W_0[0,5,10,19]$. Store $W_0[0,5,10,19] \parallel \Delta X_1[0] \parallel X_1[0]$ in H_4 indexed by $\Delta P[0,5,10,19]$. Thus, H_4 has 2^{32} rows, each of which has 2^8 values, on average.

• H_5: Guess the values of $\Delta X_1[5] \parallel X_1[4,5,6,7]$. Compute the value of $W_0[4,9,14,23] \parallel \Delta W_0[4,9,14,23]$. Store $W_0[4,9,14,23] \parallel \Delta X_1[5] \parallel X_1[5]$ in H_5 indexed by $\Delta P[4,9,14,23]$. Thus, H_5 has 2^{32} rows, each of which has 2^8 values, on average.

• H_6: Guess the values of $\Delta X_1[10] \parallel X_1[8,9,10,11]$. Compute the values of $W_0[8,13,17,23] \parallel \Delta W_0[8,13,17,23]$. Store $W_0[8,13,17,23] \parallel \Delta X_1[10] \parallel X_1[10]$ in H_6 indexed by $\Delta P[8,13,17,23]$. Thus, H_6 has 2^{32} rows, each of which has 2^8 values, on average.

• H_7: Guess the values of $\Delta X_1[19] \parallel X_1[16,17,18,19]$. Compute the values of $W_0[16,21,26,7] \parallel \Delta W_0[16,21,26,7]$. Store $W_0[16,21,26,7] \parallel \Delta X_1[19] \parallel X_1[19]$ in H_7 indexed by $\Delta P[16,21,26,7]$. Thus, H_7 has 2^{32} rows, each of which has 2^8 values, on average.

• H_8^a: Guess the values of $\Delta X_2[0] \parallel X_2[0,1,2,3]$. Compute the values of $W_1[0,5,10,19] \parallel \Delta W_1[0,5,10,19]$. Store $W_1[0,5,10,19]$ in H_8^a indexed by $\Delta W_1[0,5,10,19]$. Thus, H_8^a has 2^{32} rows, each of which has 2^8 values, on average.

• H_8^b: Guess the values of $\Delta X_2[1] \parallel X_2[0,1,2,3]$. Compute the values of $W_1[0,5,10,19] \parallel \Delta W_1[0,5,10,19]$. Store $W_1[0,5,10,19]$ in H_8^b indexed by $\Delta W_1[0,5,10,19]$. Thus, H_8^b has 2^{32} rows, each of which has 2^8 values, on average.

• H_8^c: Guess the values of $\Delta X_2[2] \parallel X_2[0,1,2,3]$. Compute the values of $W_1[0,5,10,19] \parallel \Delta W_1[0,5,10,19]$. Store $W_1[0,5,10,19]$ in H_8^c indexed by $\Delta W_1[0,5,10,19]$. Thus, H_8^c has 2^{32} rows, each of which has 2^8 values, on average.

• H_8^d: Guess the values of $\Delta X_2[3] \parallel X_2[0,1,2,3]$. Compute the values of $W_1[0,5,10,19] \parallel \Delta W_1[0,5,10,19]$. Store $W_1[0,5,10,19]$ in H_8^d indexed by $\Delta W_1[0,5,10,19]$. Thus, H_8^d has 2^{32} rows, each of which has 2^8 values, on average.

The round subkeys can be retrieved as follows.

(1) Access H_1 to compute $\mathrm{ARK}_{10}[0,15,22,25]$. Because $\mathrm{ARK}_{10}[0,15,22,25] = Z_{10}[0,15,22,25] \oplus C[0,15,22,25]$. Thus, we obtain 2^8 values of $\mathrm{ARK}_{10}[0,15,22,25]$.

(2) Access H_2 to compute $\mathrm{ARK}_{10}[11,18,21,24]$. Because $\mathrm{ARK}_{10}[11,18,21,24] = Z_{10}[11,18,21,24] \oplus C[11,18,21,24]$. Thus, there are $2^{8+8} = 2^{16}$ values of $\mathrm{ARK}_{10}[0,15,22,25] \parallel \mathrm{ARK}_{10}[11,18,21,24]$.

(3) Access H_3 to compute $\mathrm{ARK}_9^*[0,25]$. Because $\mathrm{ARK}_9^*[0,25] = Z_9[0,25] \oplus W_9^*[0,25]$, we obtain $2^{16+8} = 2^{24}$ values of $\mathrm{ARK}_{10}[0,15,22,25] \parallel \mathrm{ARK}_{10}[11,18,21,24] \parallel \mathrm{ARK}_9^*[0,25]$.

(4) Access H_4 to compute $\mathrm{ARK}_0[0,5,10,19]$. Because $\mathrm{ARK}_0[0,5,10,19] = W_0[0,5,10,19] \oplus P[0,5,10,19]$, we obtain $2^{24+8} = 2^{32}$ values of $\mathrm{ARK}_{10}[0,15,22,25] \parallel \mathrm{ARK}_{10}[11,18,21,24] \parallel \mathrm{ARK}_9^*[0,25] \parallel \mathrm{ARK}_0[0,5,10,19]$.

(5) Access H_5 to compute $\mathrm{ARK}_0[4,9,14,23]$. Because $\mathrm{ARK}_0[4,9,14,23] = W_0[4,9,14,23] \oplus P[4,9,14,23]$, we obtain $2^{32+8} = 2^{40}$ values of $\mathrm{ARK}_{10}[0,15,22,25] \parallel \mathrm{ARK}_{10}[11,18,21,24] \parallel \mathrm{ARK}_9^*[0,25] \parallel \mathrm{ARK}_0[0,5,10,19] \parallel \mathrm{ARK}_0[4,9,14,23]$.

(6) Access H_6 to compute $\mathrm{ARK}_0[8,13,17,27]$. Because $\mathrm{ARK}_0[8,13,17,27] = W_0[8,13,17,27] \oplus P[8,13,17,27]$, we obtain $2^{40+8} = 2^{48}$ values of $\mathrm{ARK}_{10}[0,15,22,25] \parallel \mathrm{ARK}_{10}[11,18,21,24] \parallel \mathrm{ARK}_9^*[0,25] \parallel \mathrm{ARK}_0[0,5,10,19] \parallel \mathrm{ARK}_0[4,9,14,23] \parallel \mathrm{ARK}_0[8,13,17,27]$.

(7) Access H_7 to compute $\mathrm{ARK}_0[16,21,26,7]$. Because $\mathrm{ARK}_0[16,21,26,7] = W_0[16,21,26,7] \oplus P[16,21,26,7]$, we obtain $2^{48+8} = 2^{56}$ values of $\mathrm{ARK}_{10}[0,15,22,25] \parallel \mathrm{ARK}_{10}[11,18,21,24] \parallel \mathrm{ARK}_9^*[0,25] \parallel \mathrm{ARK}_0[0,5,10,19] \parallel \mathrm{ARK}_0[4,9,14,23] \parallel \mathrm{ARK}_0[8,13,17,27] \parallel \mathrm{ARK}_0[16,21,26,7]$.

(8) Access H_8^a, H_8^b, H_8^c, and H_8^d to compute $\mathrm{ARK}_1[0,5,10,19]$. Because $\mathrm{ARK}_1[0,5,10,19] = W_1[0,5,10,19] \oplus X_1[0,5,10,19]$, there are 2^{10} possible values of $\mathrm{ARK}_1[0,5,10,19]$. Finally, we obtain $2^{56+10} = 2^{66}$ values of $\mathrm{ARK}_{10}[0,15,22,25] \parallel \mathrm{ARK}_{10}[11,18,21,24] \parallel \mathrm{ARK}_9^*[0,25] \parallel \mathrm{ARK}_0[0,5,10,19] \parallel \mathrm{ARK}_0[4,9,14,23] \parallel \mathrm{ARK}_0[8,13,17,27] \parallel \mathrm{ARK}_0[16,21,26,7] \parallel \mathrm{ARK}_1[0,5,10,19]$.

Thirty bytes of round subkeys are involved in our attack. For each plaintext-ciphertext pair, approximately 2^{66} possible round subkeys are removed. Thus, the probability that a wrong key is not discarded for one plaintext-ciphertext pair is approximately $1 - 2^{66-240} = 1 - 2^{-174}$. After filtering through 2^{n+95} pairs, there are approximately $2^{240} \times (1-2^{-174})^{2^{n+95}} = 1$ remaining candidates for 128 bits of the master key when $n = 86.4$. Therefore, the entire time complexity is approximately $2^{86.4+95+66} \div (7 \times 10) \approx 2^{241.3}$ encryptions, the data complexity is approximately $2^{86.4+128} = 2^{214.4}$ chosen plaintexts, and the memory complexity is approximately $2^{86.4+95} \times 4 = 2^{183.4}$ 256-bit blocks.

4.3 Impossible differential attacks on 9-round Rijndael-224-224

By appending one round at the top and two rounds at the bottom of the impossible differentials in Subsection 4.1, we mount two attacks on 9-round Rijndael-224-224. In one attack, the intermediate state X_7 by applying the MC operation on the output differences of the impossible differentials has two

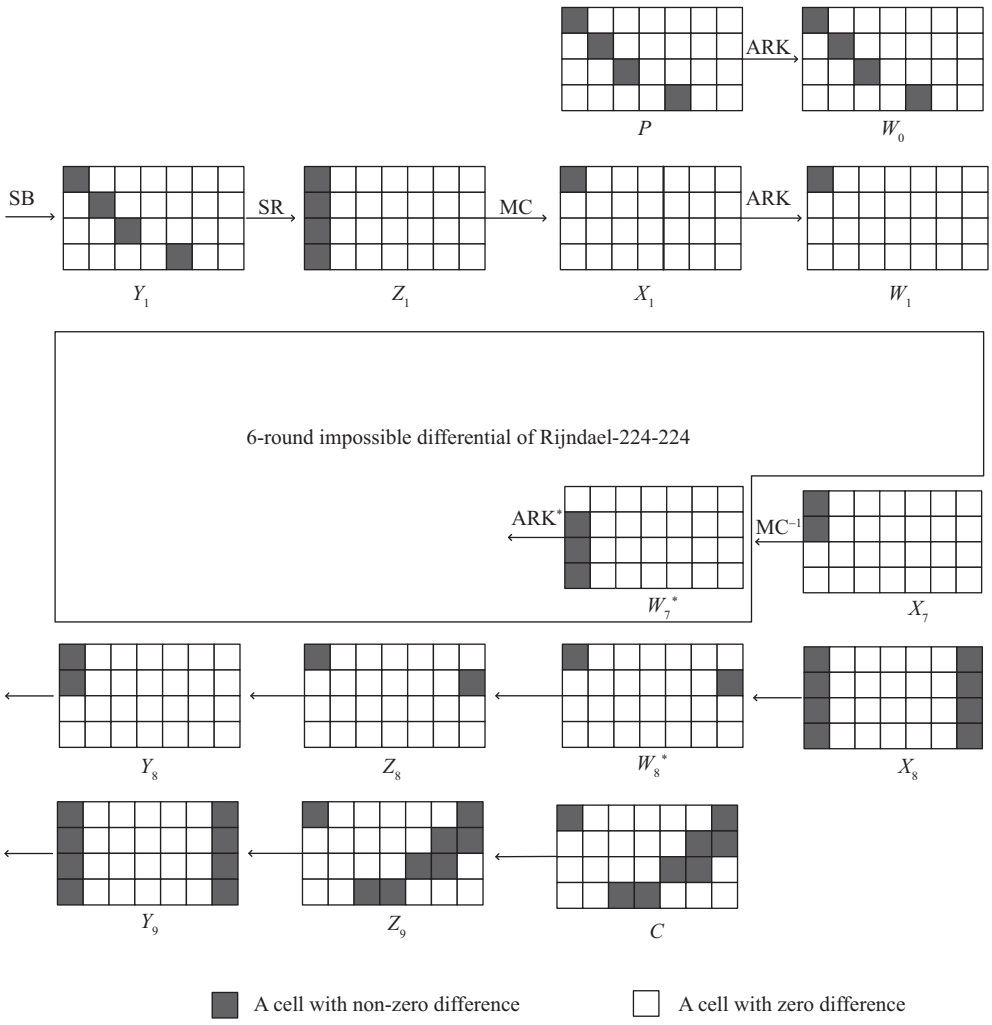

Figure 5 The first impossible differential attack on 9-round Rijndael-224-224.

non-zero bytes, as shown in Figure 5. In the other attack, the intermediate state X_7 after using the MC operation on the output differences of the impossible differentials has three non-zero bytes, as shown in Figure 6.

For the first attack, we choose 2^n structures of plaintexts. Each structure takes approximately 2^{63} pairs of plaintexts. After filtering according to the ciphertext differences, $2^{-8\times 20}\times 2^{n+63}=2^{n-97}$ pairs remain. In the key recovery phase, 14 round subkeys are involved in the analysis, which take 2^{112} possible values. Similarly, we build pre-computation tables to retrieve the round subkeys. For each plaintext-ciphertext pair, we remove 2^{34} possible round subkeys. Thus, the probability that a wrong key remains with one pair is approximately $1-2^{34-112}=1-2^{-78}$. After filtering by 2^{n-97} pairs, we have $2^{112}\times(1-2^{-78})^{2^{n-97}}=2^{-128}$ remaining candidates for $n=182.4$. Therefore, the entire time complexity is approximately $2^{182.4-97+34}\div(7\times 9)\approx 2^{113.4}$ encryptions. The data complexity is approximately $2^{182.4+32}=2^{214.4}$ chosen plaintexts. The memory complexity is approximately $2^{182.4-97}\times 4=2^{87.4}$ 224-bit blocks.

For the second attack, we choose 2^n structures of plaintexts. Each structure takes approximately 2^{63} pairs of plaintexts. We encrypt them and store the pairs that satisfy the ciphertext differences. Thus, $2^{-8\times 16}\times 2^{n+63}=2^{n-65}$ pairs remain. In the key recovery phase, 19 round-keys are involved in the analysis, which take 2^{152} possible values. For each pair, we remove approximately 2^{50} possible values by constructing precomputation tables. Thus, the probability that a wrong key is not discarded with one pair is $1-2^{50-152}=1-2^{-102}$. After filtering 2^{n-65} pairs, we have $2^{152}\times(1-2^{-102})^{2^{n-65}}=2^{-128}$

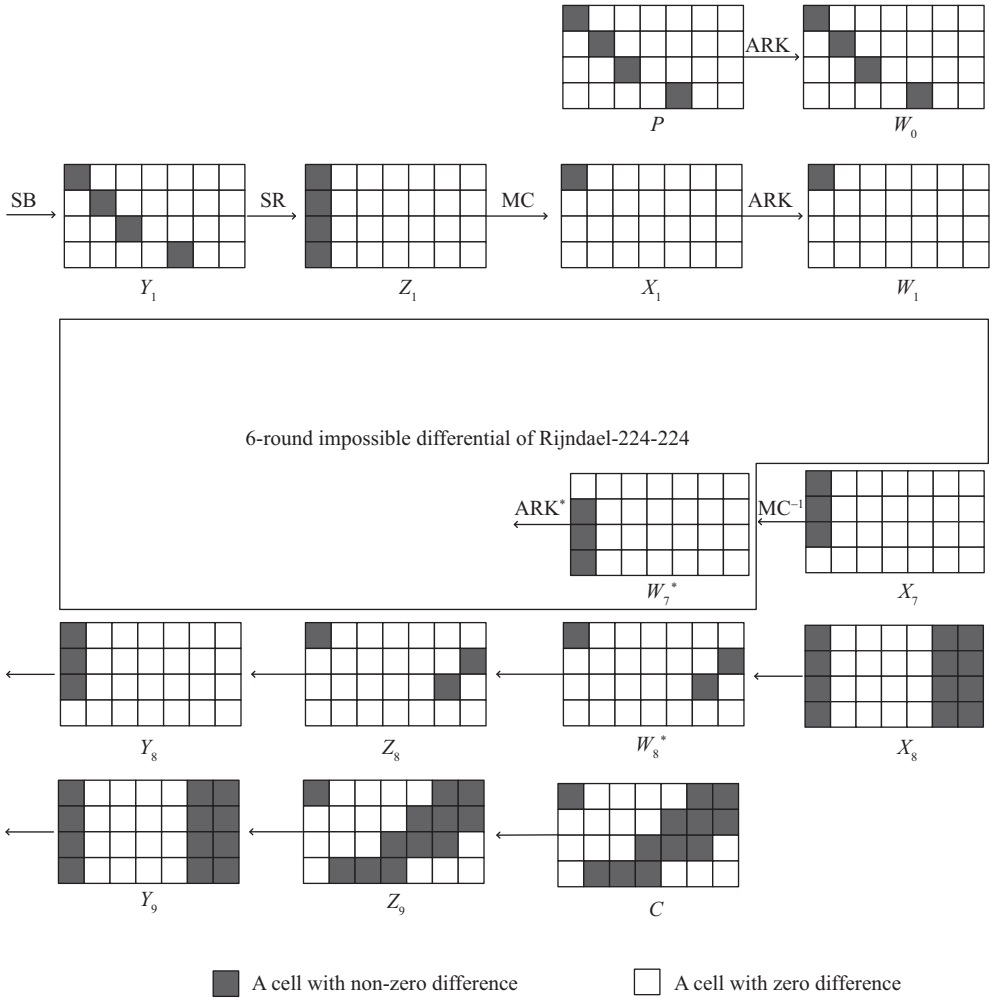

Figure 6 The second impossible differential attack on 9-round Rijndael-224-224.

remaining candidates for 32 bits of the master key when $n = 174.6$. Therefore, the entire time complexity is approximately $2^{174.6-65+50} \div (7 \times 9) \approx 2^{153.6}$ encryptions. The data complexity is approximately $2^{174.6+32} = 2^{206.6}$ chosen plaintexts. The memory complexity of the attack is approximately $2^{174.6-65} \times 4 = 2^{111.6}$ 224-bit blocks.

5 Conclusion

In this paper, we presented multiple impossible differential cryptanalyses of 10-round Rijndael-256-256, 9-round Rijndael-224-224, and 10-round Rijndael-224-256. In all attacks, we constructed precomputation tables to search the related round subkeys so that the time complexities could be reduced. For 10-round Rijndael-256-256, our attack required approximately $2^{244.4}$ chosen plaintexts, $2^{240.1}$ encryptions, and $2^{181.4}$ 256-bit blocks. For 10-round Rijndael-224-256, our attack required approximately $2^{214.4}$ chosen plaintexts, $2^{241.3}$ encryptions, and $2^{183.4}$ 224-bit blocks. For 9-round Rijndael-224-224, our attack required approximately $2^{214.4}$ chosen plaintexts, $2^{113.4}$ encryptions, and $2^{87.4}$ 224-bit blocks, or $2^{206.6}$ chosen plaintexts, $2^{153.6}$ encryptions, and $2^{111.6}$ 224-bit blocks.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61402288, 61772129, 61601292, 61672347, 61472250), Foundation of Science and Technology on Information Assurance Laboratory (Grant No. KJ-17-008), Shanghai Natural Science Foundation (Grant Nos. 15ZR1400300, 16ZR1401100), Opening Project of the Shanghai Key Laboratory of Integrated Administration Technologies for

Information Security (Grant No. AGK201703), Opening Project of the Shanghai Key Laboratory of Scalable Computing and Systems, National Cryptography Development Fund, and Fundamental Research Funds for the Central Universities.

References

1 Daemen J, Rijmen V. The design of Rijndael: AES, the advanced encryption standard. In: Information Security and Cryptography. Berlin: Springer, 2002
2 Daor J, Daemen J, Rijmen V. AES Proposal: Rijndael. http://jda.noekeon.org/, 1999
3 Phan R C W. Impossible differential cryptanalysis of 7-round Advanced Encryption Standard (AES). Inf Processing Lett, 2004, 91: 33–38
4 Biham E, Dunkelman O, Keller N. Related-key impossible differential attacks on 8-round AES-192. In: Proceedings of Cryptographers' Track at the RSA Conference — CT-RSA 2006. Berlin: Springer, 2006. 21–33
5 Biryukov A. The Boomerang attack on 5 and 6-round reduced AES. In: Proceedings of International Conference on Advanced Encryption Standard — AES 2004. Berlin: Springer, 2004. 11–15
6 Biryukov A, Khovratovich D. Related-key cryptanalysis of the full AES-192 and AES-256. In: Advances in Cryptology — ASIACRYPT 2009. Berlin: Springer, 2009. 1–18
7 Biryukov A, Khovratovich D, Nikolic I. Distinguisher and related-key attack on the full AES-256. In: Advances in Cryptology — CRYPTO 2009. Berlin: Springer, 2009. 231–249
8 Demirci H, Selçuk A A. A meet-in-the-middle attack on 8-round AES. In: Fast Software Encryption — FSE 2008. Berlin: Springer, 2008. 5086: 116–126
9 Gilbert H, Minier M. A collision attack on 7 rounds of Rijndael. In: Proceedings of the 3rd Advanced Encryption Standard Candidate Conference, New York, 2000. 230–241
10 Lu J, Dunkelman O, Keller N, et al. New impossible differential attacks on AES. In: Progress in Cryptology — INDOCRYPT 2008. Berlin: Springer, 2008. 279–293
11 Informatik T. Attacking seven rounds of Rijndael under 192-bit and 256-bit keys. In: Proceedings of the 3rd Advanced Encryption Standard Candidate Conference, New York, 2000. 215–229
12 Zhang W, Wu W, Feng D. New results on impossible differential cryptanalysis of reduced AES. In: Proceedings of International Conference on Information Security and Cryptology — ICISC 2007. Berlin: Springer, 2007. 4817: 239–250
13 Zhang W, Wu W, Zhang L, et al. Improved related-key impossible differential attacks on reduced-round AES-192. In: Selected Areas in Cryptography — SAC 2006. Berlin: Springer, 2007. 15–27
14 Biham E, Biryukov A, Shamir A. Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials. In: Advances in Cryptology — EUROCRYPT 1999. Berlin: Springer, 1999. 12–23
15 Daemen J, Knudsen L R, Rijmen V. The block cipher square. In: Fast Software Encryption — FSE 1997. Berlin: Springer, 1997. 149–165
16 Biryukov A, Shamir A. Structural cryptanalysis of SASAS. In: Advances in Cryptology — EUROCRYPT 2001. Berlin: Springer, 2001. 395–405
17 Grover L K. A fast quantum mechanical algorithm for database search. In: Proceedings of Annual ACM Symposium on the Theory of Computing — STOC 1996. New York: ACM, 1996. 24: 212–219
18 Knudsen L R. DEAL — A 128-bit block cipher. Complexity, 1998
19 Lu J, Kim J, Keller N, et al. Improving the efficiency of impossible differential cryptanalysis of reduced Camellia and MISTY1. In: Cryptographers' Track at the RSA Conference — CT-RSA 2008. Berlin: Springer, 2008. 370–386
20 Boura C, Naya-Plasencia M, Suder V. Scrutinizing and improving impossible differential attacks: applications to CLEFIA, Camellia, LBlock and Simon. In: Advances in Cryptology — ASIACRYPT 2014. Berlin: Springer, 2014. 179–199
21 Tolba M, Abdelkhalek A, Youssef A M. Impossible differential cryptanalysis of reduced-round skinny. In: Progress in Cryptology — AFRICACRYPT 2017. Cham: Springer, 2017. 117–134
22 Kim J, Hong S, Lim J. Impossible differential cryptanalysis using matrix method. Discrete Math, 2010, 310: 988–1002
23 Luo Y, Lai X, Wu Z, et al. A unified method for finding impossible differentials of block cipher structures. Inf Sci, 2014, 263: 211–220
24 Luo Y, Lai X. Improvement for finding impossible differentials of block cipher structures. IACR Cryptology ePrint Archive, 2017, 2017: 1209
25 Sasaki Y, Todo Y. New impossible differential search tool from design and cryptanalysis aspects. In: Advances in Cryptology — EUROCRYPT 2017. Cham: Springer, 2017. 185–215
26 Ding Y L, Wang X Y, Wang N, et al. Improved automatic search of impossible differentials for camellia with FL/FL^{-1} layers. Sci China Inf Sci, 2018, 61: 038103

27 Nakahara J, Pavao I C. Impossible-differential attacks on large-block Rijndael. In: Proceedings of International Conference on Information Security — ISC 2007. Berlin: Springer, 2007. 104–117
28 Zhang L, Wu W, Park J H, et al. Improved impossible differential attacks on large-block Rijndael. In: Proceedings of International Conference on Information Security — ISC 2008. Berlin: Springer, 2008. 298–315
29 Wang Q, Gu D, Rijmen V, et al. Improved impossible differential attacks on large-block Rijndael. In: Proceedings of International Conference on Information Security and Cryptology — ICISC 2012. Berlin: Springer, 2012. 298–315
30 Minier M. Improving impossible-differential attacks against Rijndael-160 and Rijndael-224. Design Code Cryptogr, 2016, 82: 1–13
31 Boura C, Minier M, Naya-Plasencia M, et al. Improved impossible differential attacks against round-reduced LBlock. Cryptogr Secur, 2014
32 Derbez P. Note on Impossible Differential Attacks. In: Fast Software Encryption — FSE 2016. Berlin: Springer, 2016. 416–427
33 Li Y J, Wu W L. Improved integral attacks on Rijndael. J Inf Sci Eng, 2011, 27: 2031–2045
34 Dunkelman O, Keller N. A new attack on the LEX stream cipher. In: Advances in Cryptology — ASIACRYPT 2008. Berlin: Springer, 2008. 539–556
35 Dunkelman O, Keller N, Shamir A. Improved single-key attacks on 8-round AES-192 and AES-256. In: Advances in Cryptology — ASIACRYPT 2010. Berlin: Springer, 2010. 158–176