ICIC Express Letters
Volume **10**, Number **5**, May **2016**

ICIC International ©2016 ISSN 1881-803X
pp. 1009–1014

RELATED-KEY ATTACK ON 8-ROUND AES-256

Jie Cui, Hong Zhong*, Runhua Shi and Yan Xu

School of Computer Science and Technology
Anhui University
No. 111, Jiulong Road, Jingkai District, Hefei 230601, P. R. China
cuijie@ahu.edu.cn; *Corresponding author: zhongh@ahu.edu.cn; cuijie@mail.ustc.edu.cn

Received November 2015; accepted February 2016

Abstract. *This paper analyzes the key schedule algorithm of AES, presents its recursion model, and describes the relationship between the expanded key bytes. According to the features of key schedule of AES-256, we propose a new 8-round attack scheme and reduce the cost of attack by changing the order of round transformation, using the alternative representation of the round keys and designing the key difference pattern properly. The time complexity is reduced from* 2^{196} *to* 2^{136}*, and the data complexity is reduced from* 2^{107} *to* 2^{72}*.*

Keywords: AES, Rijnael, Square attack, Related-key attack

1. **Introduction.** AES was announced by NIST as the Advanced Encryption Standard (AES), and AES was published as FIPS 197 in December 2001. AES is an iterated block cipher with a variable block length and a variable key length. The block length and the key length can be independently specified to 128, 192 or 256 bits. AES algorithm includes encryption, decryption and key schedule. The number of rounds depends on block length and key length [5,6]. The number of rounds is specified 10, 12 and 14 according to different combinations of block length and key length.

Now the most effective attack on AES is square attack [1,8] which was proposed by the designers of AES algorithm. Initially, the designers successfully attacked 6-round AES. The attack complexity was 2^{72} encryptions, and it required 2^{32} chosen plaintexts. J. Liu et al. [9] proposed a Lagrange interpolation attack scheme on AES-128 by combining the partial sum technique. The attacking complexity can be decreased to 2^{50}. Lucks [4] attacked 7-round AES-128/192 and 7-round AES-256. Both attacks required 2^{32} chosen plaintexts, and attack complexities were 2^{176} and 2^{192} respectively. Liu et al. [2] proposed two 7-round related-key attacks on AES-128/192. These two attacks required 2^{40} and 2^{72} chosen plaintexts respectively, and the attack complexities were 2^{184} and 2^{104} respectively. Ferguson et al. [3] proposed 8-round related-key attack on AES-256 using partial sums technology. The attack required 2^{128}–2^{119} chosen plaintexts, and the attack complexity was 2^{204}. Demirci and Selçuk [10] proposed a meet-in-the-middle attack on 8-Round AES. The attack required 2^{104} chosen plaintexts, and the time complexity was 2^{204}. Dunkelman and Keller [11] proposed three new cryptanalytic techniques, and exploited them to attack 8-round AES-256. The attack required 2^{113} chosen plaintexts, and the time complexity was 2^{196}. In 2013, P. Derbez et al. [12] proposed three new cryptanalysis schemes on 8-round AES-256. The time complexity of these three schemes was 2^{196}, and the data complexity was 2^{82}–2^{129}.

In this paper, according to the characteristics of the round transformation and the key schedule in AES-256, we propose an 8-round attack on AES-256 based on key schedule algorithm. The data complexity of our scheme is 2^{72}, and the time complexity is 2^{136}.

The rest of the paper is organized as follows. The related-key attack scheme on 8-round AES-256 is proposed in Section 2. The attack complexity analysis and comparison are given in Section 3. Finally, the paper is concluded in Section 4.

2. **Attack on 8-Round AES-256.** In this paper, the rth round input is denoted by $\mathrm{P}^{(r)}$, and plaintext is denoted by $\mathrm{P}^{(0)}$. The results of the rth round ShiftRow, MixColumn, AddRoundKey are denoted by $\mathrm{T}^{(r)}$, $\mathrm{M}^{(r)}$ and $\mathrm{B}^{(r)}$ respectively, where $0 \le r \le R$. The (i,j) byte of $\mathrm{T}^{(r)}$, $\mathrm{M}^{(r)}$, $\mathrm{B}^{(r)}$ and $\mathrm{k}^{(r)}$ is denoted by $\mathrm{t}_{i,j}^{(r)}$, $\mathrm{m}_{i,j}^{(r)}$, $\mathrm{b}_{i,j}^{(r)}$ and $\mathrm{k}_{i,j}^{(r)}$ respectively, where $0 \le i,j \le 3$.

It can be seen from above analysis that the diffusion and non-linearity of AES key schedule are not as what AES designers claimed. Ferguson et al. [3] proposed an attack on reduced AES-256 using key relevance. Liu et al. [2] proposed a 7-round related-key attack on AES-128/192. Biham et al. [7] proposed related-key rectangle attacks on reduced AES-192 and AES-256. And we also propose an 8-round attack on AES-256 using key relevance. All these successful attacks suggest that the diffusion and non-linearity of AES key schedule are not very high.

In this paper, the related-key attack is a variation of square attack. In square attack, we select a set of plaintexts satisfying that one byte or some bytes run over all possible values. While in related-key attack, we select a set of keys satisfying that one byte or some bytes run over all possible values. That is, we use 256 related keys that differ in a single byte. We use appropriate plaintext differences and key differences to cancel out the initial round and the first round key differences, and it ensures that the outputs of the first round are same. Then we track the propagation of the second round output differences (i.e., the second round key differences). When the key differences run over all possible values, after some rounds transformation, we test the balance of some bytes in the output. And we attack AES-256 according to this point.

2.1. **The key difference pattern.** With an unknown key K (either take) as a benchmark, we can obtain a set of 256 related keys $K_0, \ldots, K_{255}$. The difference $K_a \oplus K$ takes on the value a in bytes 22 and 26, and is zero elsewhere. The diffusion in the key schedule is slow enough that we can track all the differences in the round keys. For description convenience, we select two related keys K and K' from the set to discuss.

It is obvious that the difference $K \oplus K'$ is a in bytes 22 and 26, and is zero elsewhere. The key schedule for the 8-round cipher needs to generate 9 round keys. With a 128-bit block size and a 256-bit key size, this requires five cycles of the key schedule. Each of the cycles provides two round keys.

The dark gray bytes are the bytes of K that we guess. The light gray bytes are bytes that we can deduce from the guesses that we have made using the recurrence relationship between the expanded key bytes.

In the initial cycle we have a difference a in $K_{2,5}^{(0)}$ and $K_{2,6}^{(0)}$. According to the key expansion algorithm, we can get a difference a in $K_{2,5}^{(1)}$. After the second cycle key expansion, we have a difference a in $K_{2,5}^{(2)}$, $K_{2,6}^{(2)}$ and $K_{2,7}^{(2)}$. At the first half of the third cycle, the difference is first confronted with a non-linear BS transformation. To track the difference, we need to know $K_{2,7}^{(2)}$. This allows us to compute the output difference b of the BS given the input difference a. As the shading shows, this key byte can be deduced from the guesses that we have made. We thus get a difference b in $K_{1,i}^{(3)}$ for $i = 1, \ldots, 3$. At the second half of the third cycle, we also encounter a BS transformation, and therefore we need to know $K_{1,3}^{(3)}$. This gives us the output difference c of that BS given input difference b. We thus get a difference c in $K_{1,i}^{(3)}$ for $i = 4, \ldots, 7$. The differences from the previous cycle also come through as a difference a in $K_{2,5}^{(3)}$ and $K_{2,7}^{(3)}$. At the first half of the fourth cycle, we encounter two BS transformation, and therefore we need to know $K_{1,7}^{(3)}$

and $K_{2,7}^{(3)}$. This gives us the output differences d and e of the BS given input differences a and c respectively. At the second half of the fourth cycle, we also encounter two BS transformation, and therefore we need to know $K_{0,3}^{(4)}$ and $K_{1,3}^{(4)}$. This gives us the output differences f and g of the BS given input differences d and e respectively. After the fourth cycle, we have a difference e in $K_{0,i}^{(4)}$ for $i = 0, \ldots, 3$, a difference g in $K_{0,i}^{(4)}$ for $i = 4, \ldots, 7$, a difference $b \oplus d$ in $K_{1,0}^{(4)}$ and $K_{1,2}^{(4)}$, a difference $c \oplus f$ in $K_{1,4}^{(4)}$ and $K_{1,6}^{(4)}$, a difference d in $K_{1,1}^{(4)}$ and $K_{1,3}^{(4)}$, a difference f in $K_{1,5}^{(4)}$ and $K_{1,7}^{(4)}$, a difference a in $K_{2,5}^{(4)}$ and $K_{2,7}^{(4)}$. It should be noted that the latter half of the fourth cycle key expansion are not used in this attack, so $K_{0,3}^{(4)}$ and $K_{1,3}^{(4)}$ do not need to know.

In sum, if we know the four bytes: $K_{2,7}^{(2)}$, $K_{1,3}^{(3)}$, $K_{1,7}^{(3)}$ and $K_{2,7}^{(3)}$, we can obtain the key difference pattern of the five cycles key expansion after the above analysis. The key expansion process is shown in Figure 1.

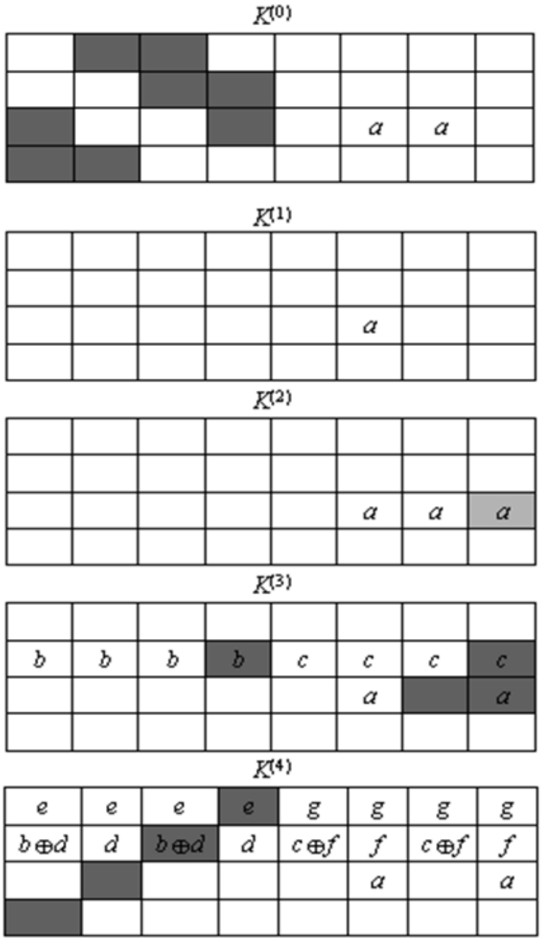

Figure 1. Difference pattern and guessing bytes in the key of the 8-round attack

2.2. **Attack process.** Having guessed the dark-gray eight bytes $(0, 1)$, $(0, 2)$, $(1, 2)$, $(1, 3)$, $(2, 0)$, $(2, 3)$, $(3, 0)$ and $(3, 1)$ of the initial round key shown in Figure 2, we encrypt one plaintext (P or P′) under each key. These plaintexts are chosen such that all encryptions end up in the same state after the first round (i.e., after adding the second round key, $\mathrm{B}^{(1)}$ and $\mathrm{B}'^{(1)}$ are same). We know the differences in the second round key $k^{(1)}$ and the key bytes that we guessed allow us to introduce appropriate differences in the plaintexts to ensure the same state after round 1. Since the round keys $k^{(2)}$ for round 2 are same, then the results of the round 2 are same. We now get one byte is different at the end of round 3; if we look at all our 256 encryptions, this one byte takes on each value exactly once. This propagates to ensure that one column bytes of $B^{(4)}$ run over all possible values when taken over the 256 encryptions. After round 5 *BS* transformation, this characteristic is maintained. After round 5 *SR* and *MC* transformation, all bytes run over all possible values when taken over the 256 encryptions. After round 5 *KA* transformation, three bytes are undecided, and the other thirteen bytes run over all possible values when taken over the 256 encryptions. After round 6 *BS* transformation, this characteristic is maintained. After round 6 *SR* and *MC* transformation, one column bytes sum to zero, and the other columns bytes are undecided. After round 6 *KA* transformation, three bytes are balanced, and the other thirteen bytes are undecided. According to this point, we select one byte $b_{3,2}^{(6)}$ for our attack (light-gray position in Figure 2 shows the diffusion process of the byte). In this paper, O, σ and X denote active byte, balanced byte and undecided byte respectively shown in Figure 2.

When difference a run over all possible values, we can get 256 related keys $K_0, \ldots, K_{255}$. According to the eight guessed key bytes, selecting 256 plaintexts $P_0, P_1, \ldots, P_{255}$ satisfying that these 256 plaintexts are selected such that the 256 encryptions end up in the same state after the first round, we encrypt these plaintexts using the corresponding

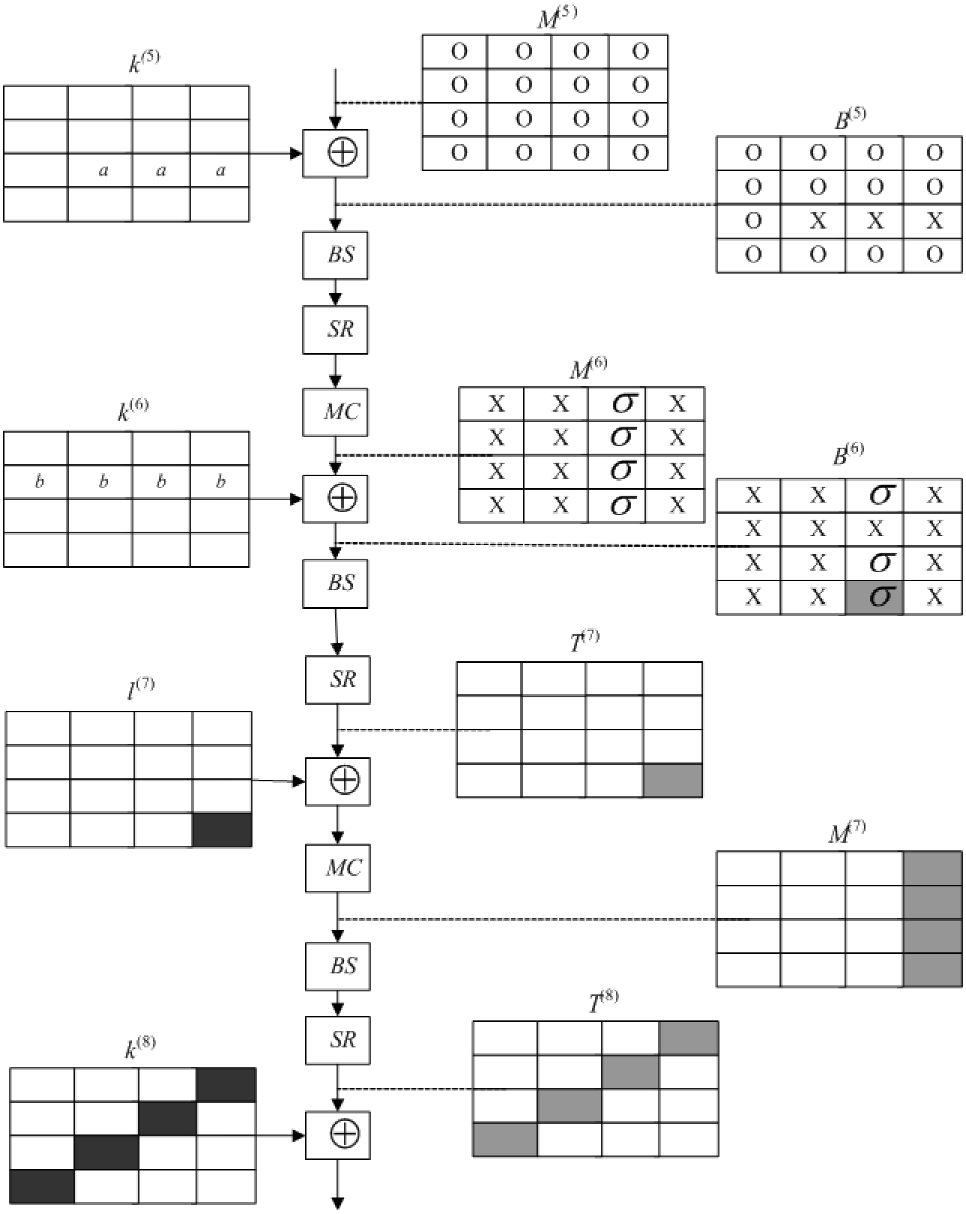

FIGURE 2. 8-round attack process of AES-256

keys and get 256 ciphertexts C_0, C_1, …, C_{255}. That is, each a determines an encryption key, then the encryption key determines a plaintext, and the encryption key and the corresponding plaintext determine an 8-round ciphertext. At the same time, each a determines a decryption key, so each a determines a $b_{3,2}^{(6)}$. According to attack process as Figure 2, we can get

$$\bigoplus_{i=0}^{255} b_{3,2}^{i(6)} = 0 \tag{1}$$

We attack 8-round AES according to Formula (1). We are going to compute $b_{3,2}^{(6)}$ from the ciphertext, our known key bytes, and some additional guessed key bytes. This is shown in Figure 2 with the gray color. Note that we are using an equivalent representation for round 7, where we have swapped the order of the MixColumn and AddRoundKey, and add $l^{(7)}$, instead of $k^{(7)}$. We need to guess one byte of $l^{(7)}$ with black color in Figure 2. The relationship between $l^{(7)}$ and $k^{(7)}$ is the following:

$$l^{(7)} = MC^{-1}\left(k^{(7)}\right)$$

According to the guessed key bytes and key difference pattern, we can get 256 keys and decrypt the corresponding ciphertexts, so we can get $b_{3,2}^{0(6)}$, $b_{3,2}^{1(6)}$, …, $b_{3,2}^{255(6)}$. We check Formula (1) whether true or false. If it is true, the guessed key is one of the candidate keys; If it is false, we have made a wrong guess somewhere, and we can generate enough sets of plaintexts to uniquely identify the correct key guesses that we have made.

3. **Attack Complexity Analysis and Comparison.** In general, we have guessed 17 bytes of key material, and for each guess we perform an amount of work comparable to a single encryption. This puts the overall complexity of the attack at 2^{136}, and we reduce the complexity greatly by contrast with the 8-round attacks [3,10-12].

Considering the plaintext requirements, we do not have to perform 256 encryptions for each of the key byte guesses that we have made. The eight bytes of plaintext that we use to cancel the differences can take on only 2^{64} values, so we can encrypt 2^{64} plaintexts with each of the 256 related keys for a total chosen plaintext requirement of 2^{72}. The attack complexity comparison results are given in Table 1.

Table 1. Comparing the complexities of the previous attacks and our attack

Cost index	The attack [3]	The attack [10]	The attack [11]	The attack [12]	Our attack
Time complexity	2^{204}	2^{204}	2^{196}	2^{196}	2^{136}
Data complexity	2^{128}–2^{119}	2^{104}	2^{113}	2^{107}	2^{72}

4. **Conclusions.** This paper analyzes the key expansion algorithm of AES, gives its recursive model, and describes the correlation between the expanded key words. According to the key relevance of AES-256, we propose a new 8-round attack scheme by changing the order of round transformation, using the alternative representation of the round keys and designing the key difference pattern properly. The time complexity is reduced from 2^{196} to 2^{136} , and the data complexity is reduced from 2^{107} to 2^{72}.

Taking into account that the number of encryption rounds of AES-256 is 14, so this study has theoretical value. This paper reveals the flaws of AES key schedule algorithm: key expansion is slow, non-linearity is too low, and key correlation is strong. This may become the starting point for further attacks in the future.

Acknowledgment. The work was supported by the National Natural Science Foundation of China (No. 61572001, No. 61502008), the Educational Commission of Anhui Province, China (No. KJ2013A017), the Natural Science Foundation of Anhui Province (No. 1508085QF132), the Research Fund for the Doctoral Program of Higher Education (No. 20133401110004), the Science and Technology Project of Anhui Province (No. 1401b042015), and the Doctoral Research Start-up Funds Project of Anhui University. The authors are very grateful to the anonymous referees for their detailed comments and suggestions regarding this paper.

REFERENCES

[1] J. Daemen and V. Rijmen, *AES Proposal: AES*, http://www.east.kuleuven.ac.be/∼rijmen/AES, 1999.

[2] J. Liu, J. Guan, Y. Liu et al., 7-round related-key attacking AES-128/192, *Journal of China Institute of Communications*, vol.24, no.6, pp.144-150, 2003.

[3] N. Ferguson, J. Kelsey, S. Lucks et al., Improved cryptanalysis of Rijndael, *Proc. of FSE 2000*, New York, USA, pp.213-230, 2001.

[4] S. Lucks, Attacking seven rounds of AES under 192-bit and 256-bit keys, *Proc. of AES Candidate Conference*, 2000.

[5] J. Chen, Y. Hu, Y. Zhang et al., Related-key square attack on AES-192, *Journal of University of Electronic Science and Technology of China*, vol.42, no.2, pp.219-224, 2013.

[6] Q. Wang, D. Gu, V. Rijmen et al., Improved impossible differential attacks on large-block Rijndael, *Pro. of ICISC*, Seoul, Korea, pp.126-140, 2012.

[7] E. Biham, O. Dunkelman and N. Keller, Related-key boomerang and rectangle attacks, *Proc. of the 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Aarhus, Denmark, pp.507-525, 2005.

[8] C.-P. Young, C.-C. Chia, Y.-B. Lin and L.-B. Chen, Fast multi-cipher transformation and its implementation for modern secure protocols, *International Journal of Innovative Computing, Information and Control*, vol.7, no.8, pp.4941-4954, 2011.

[9] J. Liu, S. Chen and L. Zhao, Lagrange interpolation attack against 6 rounds of AES-128, *Proc. of the 5th International Conference on Intelligent Networking and Collaborative Systems*, Xi'an, China, pp.652-655, 2013.

[10] H. Demirci and A. Selçuk, A meet-in-the-middle attack on 8-round AES, *Proc. of FSE*, Lausanne, Switzerland, pp.116-126, 2008.

[11] O. Dunkelman and N. Keller, A shamir improved single-key attacks on 8-round AES-192 and AES-256, *Proc. of the 16th International Conference on the Theory and Application of Cryptology and Information Security*, Singapore, pp.158-176, 2010.

[12] P. Derbez, P. Fouque and J. Jean, Improved key recovery attacks on reduced-round AES in the single-key setting, *Proc. of the 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Athens, Greece, pp.371-387, 2013.