Meet-in-the-Middle Attack on 8 Rounds of the AES Block Cipher under 192 Key Bits

Yongzhuang Wei[1,3,⋆], Jiqiang Lu[2,⋆⋆], and Yupu Hu[3]

[1] Guilin University of Electronic Technology,
Guilin City, Guangxi Province 541004, P.R. China
walker_wei@msn.com
[2] Département d'Informatique, École Normale Supérieure,
45 Rue d'Ulm, Paris 75005, France
lvjiqiang@hotmail.com
[3] State Key Laboratory of Integrated Services Networks, Xidian University,
Xi'an City, Shaanxi Province 710071, P.R. China

Abstract. The AES block cipher has a 128-bit block length and a user key of 128, 192 or 256 bits, released by NIST for data encryption in the USA; it became an ISO international standard in 2005. In 2008, Demirci and Selçuk gave a meet-in-the-middle attack on 7-round AES under 192 key bits. In 2009, Demirci et al. (incorrectly) described a new meet-in-the-middle attack on 7-round AES under 192 key bits. Subsequently, Dunkelman et al. described an attack on 8-round AES under 192 key bits by taking advantage of several advanced techniques, including one about the key schedule. In this paper, we show that by exploiting a simple observation on the key schedule, a meet-in-the-middle attack on 8-round AES under 192 key bits can be obtained from Demirci and Selçuk's and Demirci et al.'s work; and a more efficient attack can be obtained when taking into account Dunkelman et al.'s observation on the key schedule. In the single-key attack scenario, attacking 8 rounds is the best currently known cryptanalytic result for AES in terms of the numbers of attacked rounds, and our attack has a dramatically smaller data complexity than the currently known attacks on 8-round AES under 192 key bits.

Key words: Block cipher, Advanced Encryption Standard, Meet-in-middle attack.

1 Introduction

In 2001, NIST published the Advanced Encryption Standard (AES) [14] as the new-generation data encryption standard for use in the USA, designed to replace

[⋆] This author was partially supported by the Natural Science Foundation of China (No. 60833008), the Open Project Program of the State Key Laboratory of Integrated Services Networks (No. ISN11-11), and the National Basic Research 973 Program of China (No. 2007CB311201).

[⋆⋆] This author was supported by the French ANR project SAPHIR II 08-VERS-014-06. Part of his work was done when he was with Eindhoven University of Technology (Netherlands).

the Data Encryption Standard (DES) [15]. AES is a 128-bit block cipher with a user key of 128, 192 or 256 bits, which has a total of 10 rounds for a 128-bit key, 12 rounds for a 192-bit key and 14 rounds for a 256-bit key. It became a CRYPTREC-recommended e-government cipher [2] in 2002, a NESSIE selected algorithm [16] in 2003, and was adopted as an ISO international standard [11] in 2005. Since AES is increasingly widely used in many real-life cryptographic applications, it is essential to continuing to investigate its security against different cryptanalytic techniques. In this paper, we denote below by AES-128/192/256 the versions of AES that respectively use 128, 192 and 256 key bits, and we focus on the security of AES-192 in the single-key attack scenario.

Many cryptanalytic results on the security of AES-192 (in the single-key attack scenario) have been published so far [4,8,9,12,13,17,18]; and in terms of the numbers of attacked rounds, the square attack [3] on 8-round AES-192 [8] is the best currently published cryptanalytic result for AES-192, which requires almost the entire codebook and has a time complexity of 2^{188} 8-round AES encryptions.

Building on the work described in [9], in 2008 Demirci and Selçuk [4] described a 4-round property of AES, and used it as the basis of meet-in-the-middle attacks [6] on 7-round AES-192 and 8-round AES-256. In 2009, Deirci et al. [5] suggested a method to improve the 4-round property, yielding a 4-round differential [1] property, and they gave meet-in-the-middle attacks on 7-round AES-128/192 and 8-round AES-256. However, most recently Dunkelman et al. [7] pointed out a flaw in Deirci et al.'s attacks, and more importantly, Dunkelman et al. described another variant of the 4-round property due to Demirci and Selçuk, which they referred to as a multiset variant, and introduced two new cryptanalytic techniques, namely differential enumeration and key bridging, where the key bridging technique was used to drive (from AES-192's key schedule) the observation that the last column of the initial subkey can be deduced from three columns of the 8-th round key, (although they are 8 rounds away). Finally, by taking advantage of these techniques Dunkelman et al. described an attack on 8-round AES-192, which requires 2^{113} chosen plaintexts and has a time complexity of 2^{172} 8-round AES encryptions.

In this paper, we find that a meet-in-the-middle attack on 8-round AES-192 can be obtained from Demirci and Selçuk's and Demirci et al.'s work, which is based on the following two simple observations: First, we use a 4-round differential property obtained by applying Deirci et al.'s method to Demirci and Selçuk's 4-round property; and second, we observe that three concerned bytes of the 7-th round key can be deduced from the 8-th round key (this observation is not novel, and similar ones had been extensively used in previous work, for instance [12]). The attack requires 2^{36} chosen plaintexts and has a time complexity of $2^{190.63}$ 8-round AES encryptions, excluding a one-off precomputation with a time complexity of $2^{190.63}$ 8-round AES encryptions. Further, we can reduce the attack's time complexity to $2^{182.63}$ 8-round AES-192 encryptions by using Dunkelman et al.'s observation on the key schedule. Finally, with a data-time-memory tradeoff [10], we can obtain an 8-round AES-192 attack which requires 2^{41} chosen

Table 1. Cryptanalytic results on 8-round AES-192 in the single-key attack scenario

Attack Type	*Data*	*Memory*	*Time*	*Precomputation*	*Source*
Square	$2^{128} - 2^{119}$ CP	2^{64} Bytes	2^{188} Enc.	/	[8]
Meet-in-the-middle	2^{113} CP	2^{133} Bytes	2^{172} Enc.	2^{132} Enc.	[7]
	2^{36} CP	2^{193} Bytes	$2^{190.63}$ Enc.	$2^{190.63}$ Enc.	Sect. 4.2
	2^{41} CP	2^{190} Bytes	$2^{187.63}$ Enc.	$2^{187.63}$ Enc.	Sect. 4.3

plaintexts and has a time complexity of $2^{187.63}$ 8-round AES-192 encryptions. When compared with the currently known attacks on 8-round AES-192, our attack has a greater time and memory complexity, but it has a dramatically smaller data complexity. Table 1 summarises previous and our new cryptanalytic results on 8-round AES-192 in the single-key attack scenario, where CP refers to the required number of chosen plaintexts, and Enc. refers to the required number of encryption operations of 8-round AES-192.

The remainder of the paper is organised as follows. In the next section we describe the notation and the AES block cipher when used with a 192-bit key. In Section 3, we review some related results from previous work. In Section 4, we present our meet-in-the-middle attacks on 8-round AES-192. Section 5 concludes the paper.

2 Preliminaries

In this section we give the notation used throughout this paper, and then briefly describe the AES block cipher when used with a 192-bit key.

2.1 Notation

In all descriptions we assume that a number without a prefix expresses a decimal number, and a number with prefix $0x$ expresses a hexadecimal number. We use the following notation throughout this paper.

$\oplus$	bitwise logical exclusive OR (XOR) of two bit srings of the same length
$\lll$	left rotation of a bit string
$\bullet$	polynomial multiplication modulo the polynomial $x^8 + x^4 + x^3 + x + 1$ in GF(2^8)

The 16 bytes of a 4×4 byte array are numbered from top to bottom from left to right, starting with 0; an example is given below, where $a_0, a_1, \cdots, a_{15} \in \{0,1\}^8$.

$$A = (a_i)_{i=0,1,\cdots,15} = \begin{pmatrix} a_0 & a_4 & a_8 & a_{12} \\ a_1 & a_5 & a_9 & a_{13} \\ a_2 & a_6 & a_{10} & a_{14} \\ a_3 & a_7 & a_{11} & a_{15} \end{pmatrix}.$$

2.2 The AES Block Cipher

AES [14] uses the following four elementary operations to construct its round function:

- The AddRoundKey operation (denoted below by ARK) XORs a 4 × 4 byte array with a 16-byte subkey.
- The SubBytes operation (denoted below by SB) applies the same 8 × 8-bit bijective S-box (denoted below by S) 16 times in parallel to a 4 × 4 byte array.
- The ShiftRows operation (denoted below by SR) cyclically shifts the jth row of a 4 × 4 byte array to the left by j bytes, $(0 \leq j \leq 3)$.
- The MixColumns operation (denoted below by MC) pre-multiplies a 4 × 4 byte array by a fixed 4 × 4 byte matrix M. The matrix M and its inverse M^{-1} are as follows.

$$M = \begin{pmatrix} 0x02 & 0x03 & 0x01 & 0x01 \\ 0x01 & 0x02 & 0x03 & 0x01 \\ 0x01 & 0x01 & 0x02 & 0x03 \\ 0x03 & 0x01 & 0x01 & 0x02 \end{pmatrix}, \quad M^{-1} = \begin{pmatrix} 0x0e & 0x0b & 0x0d & 0x09 \\ 0x09 & 0x0e & 0x0b & 0x0d \\ 0x0d & 0x09 & 0x0e & 0x0b \\ 0x0b & 0x0d & 0x09 & 0x0e \end{pmatrix}.$$

AES-192 uses a total of thirteen 128-bit subkeys K_i, $(0 \leq i \leq 12)$, all derived from a user key K of six 32-bit words long. The key schedule is as follows, where $Rcon[i/6]$ are public constants.

```
1. Represent the user key $K$ as six 32-bit words $(W_0, W_1, ..., W_5)$.
2. For $j = 6$ to 51:
     – if $j \bmod 6 = 0$ then $W_j = W_{j-6} \oplus \mathrm{SB}(W_{j-1} \lll 8) \oplus Rcon[j/6]$;
     – else $W_j = W_{j-6} \oplus W_{j-1}$.
3. $K_i = (W_{4i}, W_{4i+1}, W_{4i+2}, W_{4i+3})$, $(0 \leq i \leq 12)$.
```

AES takes as input a 128-bit plaintext block P, represented as a 4 × 4 byte array. AES-192 has a total of 12 rounds, and its encryption procedure is follows, where x is a 16-byte variable.

```
1. $x = \mathrm{ARK}(P, K_0)$.
2. For $i = 1$ to 11:
     $x = \mathrm{SB}(x)$,
     $x = \mathrm{SR}(x)$,
     $x = \mathrm{MC}(x)$,
     $x = \mathrm{ARK}(x, K_i)$.
3. $x = \mathrm{SB}(x), x = \mathrm{SR}(x)$.
4. Ciphertext = $\mathrm{ARK}(x, K_{12})$.
```

An equivalent description of the algorithm can be derived by reversing the order of the third and fourth operations of Step 2 of the above description, i.e. the operations involving MC and ARK. These two steps then become:

$x = \mathrm{ARK}(x, \widehat{K}_i)$,

$x = \mathrm{MC}(x)$,

where $\widehat{K}_i = \mathrm{MC}^{-1}(K_i)$. We use this alternative representation in certain of the attacks described later.

The ith iteration of Step 2 in the above description is referred to below as Round i, and the transformations in Steps 3 and 4 are referred to below as the final round (i.e. Round 12). We write $K_{i,j}$ (respectively, $\widehat{K}_{i,j}$) for the jth byte of K_i (respectively, $\widehat{K}_i$), $(0 \leq j \leq 15)$.

3 Related Results from Previous Work

In this section, we briefly review some related results from Demirci and Selçuk's, Demirci et al.'s and Dunkelman et al.'s work, which will be used in our attack. We refer the reader to [4, 5, 7] for details.

3.1 Demirci and Selçuk's Attack on 7-Round AES-192

In 2008, Demirci and Selçuk [4] described the following 4-round property for AES.

Proposition 1 (A 4-Round Property). *Let $\mathcal{S}$ be a set of 256 4×4 byte arrays $X^{(i)} = (x_j^{(i)})_{j=0,1,\cdots,15}$ with byte (0) taking all the possible values and the other 15 bytes fixed, $(i = 0, 1, \cdots, 255)$. If $Y^{(i)} = (y_j^{(i)})_{j=0,1,\cdots,15}$ is the result of encrypting $X^{(i)}$ using 4 rounds of AES, then $y_0^{(i)}$ can be expressed with a function of $x_0^{(i)}$ and 25 constant 8-bit parameters $c_0, c_1, \cdots, c_{24}$, written $y_0^{(i)} = f_{c_0,c_1,\cdots,c_{24}}(x_0^{(i)})$.*

Building on the 4-round property, Demirci and Selçuk first gave a basic meet-in-the-middle attack on 7-round AES; the attack procedure can be described as follows.

1. For each of the $2^{25\times8} = 2^{200}$ possible values of the 25 parameters $c_0, c_1, \cdots$, c_{24}, precompute $f_{c_0,c_1,\cdots,c_{24}}(x)$ sequentially for $x = 0, 1, \cdots$, 255. Store the 2^{200} 256-byte sequences in a hash table.
2. Choose a set of 2^{32} plaintexts with bytes (0,5,10,15) of the 2^{32} plaintexts taking all the possible values and the other 12 bytes fixed. In a chosen-plaintext attack scenario, obtain the corresponding ciphertexts.
3. Guess a value for $(K_{0,0}, K_{0,5}, K_{0,10}, K_{0,15}, K_{1,0})$, and then do as follows.
   (a) Partially encrypt the set of 2^{32} plaintexts with the guessed $(K_{0,0}, K_{0,5}$, $K_{0,10}, K_{0,15}, K_{1,0})$ to get the intermediate values for byte (0) just after the first round.
   (b) Choose 256 plaintexts such that the intermediate values for byte (0) just after the first round distribute uniformly among $[0, 1, \cdots, 255]$ and the intermediate values for the other bytes just after the first round are constant.
   (c) Sort the 256 plaintexts chosen in Step 3(b) in the sequence indexed by their values in byte (0) just after the first round.

4. Guess a value for $(\widehat{K}_{6,0}, K_{7,0}, K_{7,7}, K_{7,10}, K_{7,13})$, and then partially decrypt the sequence of ciphertexts corresponding to the sequence of 256 plaintexts obtained in Step 3(c) with the guessed $(K_{7,0}, K_{7,7}, K_{7,10}, K_{7,13}, \widehat{K}_{6,0})$ to get the sequence of the intermediate values for byte (0) just before the sixth round. Compare this sequence with each of the 2^{200} sequences obtained in Step 1; if it matches one of them, record the guessed value for $(K_{0,0}, K_{0,5}, K_{0,10}, K_{0,15}, K_{1,0}, \widehat{K}_{6,0}, K_{7,0}, K_{7,7}, K_{7,10}, K_{7,13})$ and execute Step 5; otherwise, repeat Steps 3 and 4 with another guess.
5. Execute similarly Steps 1–4 with $y_0^{(i)}$ being replaced by $y_1^{(i)}, y_2^{(i)}, y_3^{(i)}$ in turn, and finally obtain $(\widehat{K}_{6,1}, \widehat{K}_{6,2}, \widehat{K}_{6,3}, K_{7,1}, \cdots, K_{7,6}, K_{7,8}, K_{7,9}, K_{7,11}, K_{7,12}, K_{7,14}, K_{7,15})$.
6. Exhaustively search the remaining key bytes.

The precomputation has a time complexity of approximately $256 \times 2^{200} \times 1.5 \times \frac{1}{7} \approx 2^{205.78}$ 7-round AES encryptions under the rough estimate that a computation of $f_{c_0,c_1,\cdots,c_{24}}$ equals 1.5 one-round AES encryption in terms of time. The attack requires 2^{32} chosen plaintexts and a memory of 2^{210} bytes; its time complexity is dominated by that for executing Step 4 four times, and it has a time complexity of approximately $256 \times 2^{8\times 10} \times \frac{2}{4\times 7} \times 4 \approx 2^{86.2}$ 7-round AES encryptions, where $\frac{2}{4\times 7}$ represents the ratio of the number of the columns that need to decrypt to the total number of the columns in 7-round AES.

Finally, Demirci and Selçuk described a data-time-memory tradeoff version of the above basic attack which can be applied to 7-round AES-192 (for some $n > 14$): The precomputation has a time complexity of $2^{205.78-n}$ 7-round AES encryptions, and with a success probability of 98%, the attack requires 2^{34+n} chosen plaintexts and a memory of $2^{210-4\times n}$ bytes, and has a time complexity of $2^{88.2+n}$ 7-round AES encryptions.

3.2 Demirci et al.'s Method to Improve the 4-Round Property

Observe that there are 25 constant parameters for f in Demirci and Selçuk's 4-round property. It would be desirable to decrease the number of parameters.

In 2009, Demirci et al. [5] suggested the following method to improve Demirci and Selçuk's 4-round property: Consider the difference between the result of encrypting $X^{(i)}$ using 4 rounds of AES and the result of encrypting another byte array $X^{(l)} = (x_j^{(l)})_{j=0,1,\cdots,15}$ from the set $\mathcal{S}$ using 4 rounds of AES, that is $y_0^{(i)} \oplus y_0^{(l)} = f_{c_0,c_1,\cdots,c_{24}}(x_0^{(i)}) \oplus f_{c_0,c_1,\cdots,c_{24}}(x_0^{(l)})$. By this method, one constant parameter (i.e., the first byte of the 4-th round key) is canceled out. We refer to a 4-round property using this method as a 4-round differential property.

If we apply Demirci et al.'s method to Demirci and Selçuk's 4-round property, then we can easily get a 4-round differential property with 24 constant parameters. Demirci et al. did not describe this 4-round differential property in their paper, but instead they gave a 4-round differential property with 15 constant parameters that holds with probability 2^{-72}, and finally used it to conduct meet-in-the-middle attacks on 7-round AES-128/192 and 8-round AES-256; the attack procedures are similar to Demirci and Selçuk's attacks, and the main difference is due to use of the 4-round differential property with 15 constant parameters. Besides, it is worthy to note that Demirci et al. computed $y_0^{(i)} \oplus y_0^{(l)}$ only for 32 pairs of $(x_0^{(i)}, x_0^{(l)})$, where $x_0^{(i)}$ is fixed to 0 and $x_0^{(l)}$ ranges from 1 to 32. However, Dunkelman et al. [7] found recently that the time complexities of Demirci et al.'s attacks are highly underestimated.

3.3 An Observation on the Key Schedule due to Dunkelman et al.

In [7], Dunkelman et al. introduced another variant of Demirci and Selçuk's 4-round property, which looks similar to but rather different in nature from the 4-round differential property obtained by applying Demirci et al.'s method to Demirci and Selçuk's 4-round property. It yields an attack on 8-round AES-192, together with two other cryptanalytic techniques. Here we are only interested in their novel observation on the key schedule of AES-192, as follows.

Proposition 2. *The subkey bytes* $(K_{0,12}, K_{0,13}, K_{0,14}, K_{0,15})$ *can de deduced from the subkey bytes* $(K_{8,0}, \cdots, K_{8,7}, K_{8,12}, \cdots, K_{8,15})$.

4 Meet-in-the-Middle Attack on 8-Round AES-192

In this section, we show that by exploiting a simple observation on the key schedule, a meet-in-the-middle attack on 8-round AES-192 can be obtained based on Demirci and Selçuk's and Demirci et al.'s work. Finally, we improve the attack following Dunkelman et al.'s observation described in Proposition 2.

4.1 Preliminary Results

First, by the key schedule of AES-192, we easily get the following equations:

$$\begin{aligned}\widehat{K}_{7,3} = {} & 0x0b \bullet (K_{8,4} \oplus K_{8,8}) \oplus 0x0d \bullet (K_{8,5} \oplus K_{8,9}) \oplus \\ & 0x09 \bullet (K_{8,6} \oplus K_{8,10}) \oplus 0x0e \bullet (K_{8,7} \oplus K_{8,11}); \qquad (1)\\ \widehat{K}_{7,6} = {} & 0x0d \bullet (K_{8,8} \oplus K_{8,12}) \oplus 0x09 \bullet (K_{8,9} \oplus K_{8,13}) \oplus \\ & 0x0e \bullet (K_{8,10} \oplus K_{8,14}) \oplus 0x0b \bullet (K_{8,11} \oplus K_{8,15}); \qquad (2)\\ \widehat{K}_{6,12} = {} & 0x0e \bullet (K_{8,0} \oplus K_{8,4}) \oplus 0x0b \bullet (K_{8,1} \oplus K_{8,5}) \oplus \\ & 0x0d \bullet (K_{8,2} \oplus K_{8,6}) \oplus 0x09 \bullet (K_{8,3} \oplus K_{8,7}). \qquad (3)\end{aligned}$$

Next, we can similarly obtain the following 4-round differential property by applying Demirci et al.'s method to Demirci and Selçuk's 4-round property. The reason that we target byte (12) is that we can deduce byte (12) of $\widehat{K}_6$ (i.e. $\widehat{K}_{6,12}$) from the 8-th round key K_8 by Eq. (3).

Proposition 3 (A 4-Round Differential Property). *Consider a set of 256* 4×4 *byte arrays* $X^{(i)} = (x_j^{(i)})_{j=0,1,\cdots,15}$ *with byte (12) taking all the possible* *values and the other 15 bytes fixed, $(i = 0, 1, \cdots, 255)$. If $Y^{(i)} = (y_j^{(i)})_{j=0,1,\cdots,15}$ is the result of encrypting $X^{(i)}$ using 4 rounds of AES, then $y_{12}^{(i)} \oplus y_{12}^{(m)}$ can be expressed with a function of $x_{12}^{(i)}, x_{12}^{(m)}$ and 24 constant 8-bit parameters $c'_0, c'_1, \cdots, c'_{23}$, written $y_{12}^{(i)} \oplus y_{12}^{(m)} = g_{c'_0,c'_1,\cdots,c'_{23}}(x_{12}^{(i)}, x_{12}^{(m)})$, where $m \in [0, 255]$.*

4.2 Attacking 8-Round AES-192

Using the 4-round differential property given in Proposition 3, we can now devise a meet-in-the-middle attack on 8-round AES-192; the attack is solely based on Demirci and Selçuk's and Demirci et al.'s work and the above observation on the key schedule, and its procedure is as follows, where n_1 and n_2 are small non-negative numbers and their specific values will be given below.

1. For each of 2^{192-n_1} possible values of the 24 parameters $c'_0, c'_1, \cdots, c'_{23}$, precompute $g_{c'_0,c'_1,\cdots,c'_{23}}(0, x)$ sequentially for $x = 1, 2, \cdots, 32$. Store the 2^{192-n_1} 32-byte sequences in a hash table $\mathcal{L}$.
2. Choose 2^{n_2} structures S_i, $(i = 0, 1, \cdots, 2^{n_2} - 1)$, where a structure S_i is defined to be a set of 2^{32} plaintexts $P_{i,j}$ with bytes (1,6,11,12) of the 2^{32} plaintexts taking all the possible values and the other 12 bytes fixed, $(j = 0, 1, \cdots, 2^{32}-1)$. In a chosen-plaintext attack scenario, obtain the ciphertexts for the 2^{n_2} structures of 2^{32} plaintexts; we denote by $C_{i,j}$ the ciphertext for plaintext $P_{i,j}$.
3. Guess a value for $(K_{0,1}, K_{0,6}, K_{0,11}, K_{0,12}, K_{1,12})$, and then do as follows for each structure S_i.
   (a) Partially encrypt the set of 2^{32} plaintexts $P_{i,j}$ with the guessed $(K_{0,1}, K_{0,6}, K_{0,11}, K_{0,12}, K_{1,12})$ to get the intermediate values for byte (12) just after the first round.
   (b) Choose 33 plaintexts such that the intermediate values for byte (12) just after the first round distribute uniformly among $[0, 1, \cdots, 32]$ and the intermediate values for the other bytes just after the first round are constant. Sort them in the sequence indexed by their values in byte (12) just after the first round; and we denote it by $(\widehat{P}_{i,0}, \widehat{P}_{i,1}, \cdots, \widehat{P}_{i,32})$.
   (c) Guess a value for $(K_8, \widehat{K}_{7,9}, \widehat{K}_{7,12})$, and do as follows.
      i. Compute $(\widehat{K}_{6,12}, \widehat{K}_{7,3}, \widehat{K}_{7,6})$ by Eqs. (1)–(3).
      ii. Partially decrypt the sequence of ciphertexts corresponding to $(\widehat{P}_{i,0}, \widehat{P}_{i,1}, \cdots, \widehat{P}_{i,32})$ with $(K_8, \widehat{K}_{7,3}, \widehat{K}_{7,6}, \widehat{K}_{7,9}, \widehat{K}_{7,12}, \widehat{K}_{6,12})$ to get the sequence of the intermediate values for byte (12) just before the sixth round; and we denote it by $(T_{i,0}, T_{i,1}, \cdots, T_{i,32})$.
      iii. Compute $(T_{i,0} \oplus T_{i,1}, T_{i,0} \oplus T_{i,2}, \cdots, T_{i,0} \oplus T_{i,32})$, and then check whether this sequence matches a sequence in $\mathcal{L}$; if so, record the guessed value for $(K_{0,1}, K_{0,6}, K_{0,11}, K_{0,12}, K_{1,12}, \widehat{K}_{7,9}, \widehat{K}_{7,12}, K_8)$ and execute Step 4; otherwise, repeat Step 3 with another structure of plaintexts (or another subkey guess when all the 2^{n_2} structures are tested).

4. For every recorded value for $(\widehat{K}_{7,9}, \widehat{K}_{7,12}, K_8)$, exhaustively search the remaining key bytes.

The attack requires 2^{32+n_2} chosen plaintexts. The one-off precomputation has a time complexity of $33 \times 2^{192-n_1} \times 1.5 \times \frac{1}{8} \approx 2^{194.63-n_1}$ 8-round AES-192 encryptions. The time complexity of Step 3(a) is $2^{32+n_2} \times 2^{40} \times \frac{1}{4\times 8} = 2^{67+n_2}$ 8-round AES-192 encryptions, where $\frac{1}{4\times 8}$ represents the ratio of the number of the columns that need to encrypt to the total number of the columns in 8-round AES. The time complexity of Step 3(c) is dominated by the time complexity of Step 3(c)-(ii), which is $2^{n_2} \times 33 \times 2^{40+144} \times \frac{6}{4\times 8} \approx 2^{186.63+n_2}$ 8-round AES-192 encryptions, where $\frac{6}{4\times 8}$ represents the ratio of the number of the columns that need to decrypt to the total number of the columns in 8-round AES. In Step 3(c)-(iii), for a wrong guess of $(K_{0,1}, K_{0,6}, K_{0,11}, K_{0,12}, K_{1,12}, \widehat{K}_{7,9}, \widehat{K}_{7,12}, K_8)$, the probability that the sequence $(T_{i,0} \oplus T_{i,1}, T_{i,0} \oplus T_{i,2}, \cdots, T_{i,0} \oplus T_{i,32})$ matches a sequence in $\mathcal{L}$ is approximately $1 - \binom{2^{192-n_1}}{0}(2^{-32\times 8})^0(1-2^{-32\times 8})^{2^{192-n_1}} \approx 2^{-32\times 8} \times 2^{192-n_1} = 2^{-64-n_1}$, and thus the probability that a sequence from the set $\{(T_{i,0}\oplus T_{i,1}, T_{i,0}\oplus T_{i,2}, \cdots, T_{i,0}\oplus T_{i,32})|i = 0, 1, \cdots, 2^{n_2}-1\}$ matches a sequence in $\mathcal{L}$ is approximately $1-\binom{2^{n_2}}{0}(2^{-64-n_1})^0(1-2^{-64-n_1})^{2^{n_2}} \approx 2^{-64-n_1+n_2}$, (assuming both the events have a binomial distribution). Consequently, it is expected that about $2^{40+144} \times 2^{-64-n_1+n_2} = 2^{120-n_1+n_2}$ values for $(K_{0,1}, K_{0,6}, K_{0,11}, K_{0,12}, K_{1,12}, \widehat{K}_{7,9}, \widehat{K}_{7,12}, K_8)$ are recorded in Step 3(c)-(iii). As a result, Step 4 takes at most $2^{120-n_1+n_2} \times 2^{48} = 2^{168-n_1+n_2}$ 8-round AES-192 encryptions.

In Step 3(c)-(iii), for the correct guess of $(K_{0,1}, K_{0,6}, K_{0,11}, K_{0,12}, K_{1,12}, \widehat{K}_{7,9}, \widehat{K}_{7,12}, K_8)$, the probability that a sequence from the set $\{(T_{i,0} \oplus T_{i,1}, T_{i,0} \oplus T_{i,2}, \cdots, T_{i,0} \oplus T_{i,32})|i = 0, 1, \cdots, 2^{n_2} - 1\}$ matches a sequence in $\mathcal{L}$ is $1 - \binom{2^{n_2}}{0}(\frac{2^{192-n_1}}{2^{192}})^0(1-\frac{2^{192-n_1}}{2^{192}})^{2^{n_2}} = 1-(1-\frac{1}{2^{n_1}})^{2^{n_2}}$.

Let $n_1 = n_2 = 4$, then the one-off precomputation has a time complexity of $2^{190.63}$ 8-round AES-192 encryptions, and the attack requires 2^{36} chosen plaintexts and a memory of 2^{193} bytes, and has a time complexity of $2^{190.63}$ 8-round AES-192 encryptions. The attack has a success probability of $1-(1-\frac{1}{2^4})^{2^4} \approx$ 65%.

Notes:

1. As mentioned in [10], the time complexity of a one-off precomputation is typically not counted as part of the time complexity of an attack, since it can be performed at the cryptanalyst's leisure. We notice that this might be controversial, and for conservatism we make the sum of all the time complexities in each of our attacks smaller than that for exhaustive key search.
2. Observe that meet-in-the-middle attacks on 8-round AES-256 can be easily obtained by modifying the above 8-round AES-192 attack procedure. A typical one requires 2^{32} chosen plaintexts and a memory of $2^{197.33}$ bytes, and has a time complexity of $2^{202.95}$ 8-round AES-256 encryptions, plus a precomputation that has a time complexity of $2^{194.95}$ 8-round AES-256 encryptions. This is slightly better than but comparable to the 8-round AES-256 attack presented in [4].

4.3 Improving the 8-Round AES-192 Attack

By Dunkelman et al.'s proof for the observation described in Proposition 2, we have the following equation, where θ represents the first byte of $Rcon[4]$.

$$K_{8,0} = K_{0,12} \oplus K_{8,4} \oplus S(K_{8,5} \oplus K_{8,13}) \oplus \theta.$$

Thus, we do not need to guess $K_{8,0}$ in Step 3(c) of the above 8-round AES-192 attack, reducing the attack's time complexity by a factor of 2^8. Further, we can obtain a data-time-memory tradeoff version with a success probability of 98%: The precomputation has a time complexity of $2^{194.63-n_1}$ 8-round AES-192 encryptions, and the attack requires 2^{34+n_2} chosen plaintexts and a memory of 2^{197-n_1} bytes, and has a time complexity of $2^{180.63+n_2}$ 8-round AES-192 encryptions. Typically, let $n_1 = n_2 = 7$, then the precomputation has a time complexity of $2^{187.63}$ 8-round AES-192 encryptions, and the attack requires 2^{41} chosen plaintexts and a memory of 2^{190} bytes, and has a time complexity of $2^{187.63}$ 8-round AES-192 encryptions.

5 Conclusion

We have given a meet-in-the-middle attack on 8-round AES-192, building solely on Demirci and Selçuk's and Demirci et al.'s work [4,5] plus a simple observation on the key schedule. Finally, we have described a more efficient attack which is based on encrypting 2^{41} chosen plaintexts and has a time complexity of $2^{187.63}$ 8-round AES-192 encryptions. Our attack has a greater time and memory complexity than the currently known attacks on 8-round AES-192, however its data complexity is dramatically smaller.

Acknowledgments

The authors are very grateful to the anonymous referees for their comments.

References

1. Biham, E., Shamir, A.: Differential cryptanalysis of the Data Encryption Standard. Springer (1993)
2. CRYPTREC — Cryptography Research and Evaluation Committees, report 2002.
3. Daemen, J., Knudsen, L.R., Rijmen, V.: The block cipher Square. In: FSE 1997. Lecture Notes in Computer Science, vol. 1267, pp. 149–165. Springer, Heidelberg (1997)
4. Demirci, H., Selçuk, A. A.: A meet-in-the-middle attack on 8-round AES. In: Nyberg, K. (ed.) FSE 2008. LNCS, vol. 5086, pp. 116–126. Springer, Heidelberg (2008)
5. Demirci, H., Taşkm, I., Çoban, M., Baysal, A.: Improved meet-in-the-middle attacks on AES. In: INDOCRYPT 2009. LNCS, vol. 5922, pp. 144-156. Springer, Heidelberg (2009)

6. Diffie, W., Hellman, M.: Exhaustive cryptanalysis of the NBS data encryption standard. Computer 10(6), pp. 74–84. IEEE (1977)
7. Dunkelman, O., Keller, N., Shamir, A.: Improved single-key attacks on 8-round AES. Cryptology ePrint Archive, Report 2010/322. Available at `http://eprint.iacr.org/2010/322.pdf`
8. Ferguson, N., Kelsey, J., Lucks, S., Schneier, B., Stay, M., Wagner, D., Whiting, D.: Improved cryptanalysis of Rijndael. In: FSE 2000. Lecture Notes in Computer Science, vol. 1978, pp. 213–230. Springer, Heidelberg (2001)
9. Gilbert, H., Minier, M.: A collision attack on 7 rounds of Rijndael. In: The Third Advanced Encryption Standard Candidate Conference, pp. 230–241. NIST (2000)
10. Hellman, M.E.: A cryptanalytic time-memory-tradeoff. IEEE Trans. Information Theory, 26(4), 401–406 (1980)
11. International Standardization of Organization (ISO), International Standard – ISO/IEC 18033-3, Information technology – Security techniques – Encryption algorithms – Part 3: Block ciphers (2005)
12. Lu, J., Dunkelman, O., Keller, N., Kim, J.: New impossible differential attacks on AES, In: INDOCRYPT 2008. Lecture Notes in Computer Science, vol. 5365, pp. 279–293. Springer, Heidelberg (2008)
13. Lucks, S.: Attacking seven rounds of Rijndael under 192-bit and 256-bit keys. In: The Third Advanced Encryption Standard Candidate Conference, pp. 215–229. NIST (2000)
14. National Institute of Standards and Technology (NIST). Advanced Encryption Standard (AES), FIPS-197 (2001)
15. National Institute of Standards and Technology (NIST). Data Encryption Standards (DES), FIPS-46 (1977)
16. NESSIE — New European Schemes for Signatures, Integrity, and Encryption, final report of European project IST-1999-12324.
17. Phan, R.: Impossible differential cryptanalysis of 7-round Advanced Encryption Standard (AES). Information Processing Letters 91(1), 33–38 (2004)
18. Zhang, W., Wu, W., Feng, D.: New results on impossible differential cryptanalysis of reduced AES. In: ICISC 2007. Lecture Notes in Computer Science, vol. 4817, pp. 239–250. Springer, Heidelberg (2007)