International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 9, No.4 (July-2020)

http://dx.doi.org/10.12785/ijcds/090422

New Results in Reduced Round AES - 256 Impossible Differential Cryptanalysis

Jithendra.K.B[1] and ShahanaT.Kassim[2]

[1] *Department of Electronics and Communication, College of Engineering, Thalassery, Kerala, India*
[2]*Division of Electronics, School of Engineering, CUSAT, Kerala, India*

Received 25 Sep.2019, Revised 31 Jan. 2020, Accepted 30 May 2020, Published 1 Jul. 2020

Abstract: Security of Crypto systems is usually analyzed through different cryptanalytic methods. Since Advance Encryption Standard (AES) is one of the most widely used and popular block cipher, a number of attacks have already been proposed on it. Lots of reduced round attacks on AES are available in the literature. In this paper, two efficient reduced round impossible differential attacks are introduced against AES - 256.The attacks proposed here show how an attack can modified for betterment. The first one is a new 8th round attack, which shows the data complexity and time complexity can get interchanged without affecting the memory requirement, by introducing proper change in the attack procedure.The second cryptanalysis is carried out in which four round impossible differential begins from third round only, wherein conventional attacks it starts from second round itself. This difference in attack procedure leads to reduction in data as well as time complexities. Moreover, the interchange of Add Round Key and Mix Column operations done in the 7th round of conventional impossible attacks can be avoided here. A conventional attack appeared in the literature is taken as the main reference. Comparison of the complexities is also given.

Keywords:AES-256, Cryptanalysis, Impossible Differentials, Complexity

1. INTRODUCTION

AES [1] has 3 versions based on the key length, which are AES - 128, AES – 192 and AES – 256 with key lengths 128, 192 and 256 bits respectively. Security of a system has a direct relation with its key length. Different types of attacks are experimented by researchers on all the three versions of reduced round AES. In this paper the cryptanalysis proposed by W. Zhang et al. [2] on AES - 256 using impossible differential cryptanalysis is modified in two different ways

Impossible differential cryptanalysis [3] searches for the non-existing differentials for elimination of wrong keys from a pool of possible keys so that only right key elements remain left. Several impossible attacks can be seen in the literature. Biham and Keller presented the 4 round impossible differential in [4] to attack 5 round AES-128. This attack is extended in [5] by Cheon J.H et al. to 6th round of AES 128. Later Raphel.C and W Phan [6] introduced some attacks on AES – 192 and AES – 256 exploiting the weakness of AES key scheduling. Obviously, attacks given in [4] and [5] are applicable to AES – 192 and AES – 256 too, since the key scheduling is not exploited. Articles [7] and [8] give improved results in reduced round AES cryptanalysis. Later in 2011 Bogdanov et al. applied biclique attack on full round AES [9]. An improved version of this attack is proposed in [10]. Cache attacks and collision attacks are illustrated in [11] and [12] respectively. Enhancement of AES security against modern attacks using variable key block ciphers are is given in [13]. Many articles were published about fault based attacks and side channel attacks on AES [14–21]. An improved version of related key impossible differential attack on AES-192 is given in [22]. Now the research on light weight block cipher has become a hot research topic in cryptography. Many light weight block ciphers have been proposed recently. Advanced methods like Mixed Integer Linear Programming (MILP) etc. are used in cryptanalysis [23, 24]. But these techniques are not practical for cryptanalysis of block ciphers constituted with 8 bit substitution boxes. At the same time, differential cryptanalysis methods still stand relevant for the analysis of any type of block ciphers.

In this paper two 8 round attacks on AES – 256 are proposed. In the first attack, the required time complexity

E-mail address: jithendrakb@gmail.com, shahanatk@cusat.ac.in

is reduced to a great extent but at the expense of data complexity. This change will be required, where time is crucial. Another 8 round attack on AES – 256 is also proposed in this paper. In conventional impossible differential cryptanalysis, the four round impossible differential begin from second round onwards. For the attack given in this paper, impossible differential begin from third round only. So the backward analysis, from the last round, is to reach only up to the third round here, but to the second round in conventional attacks. The data arrangement in the last round of *r* round conventional impossible differential attack can be used for *r+1* round, when proposed impossible differential attack is applied. We introduce this attack, taking the attack given in [2] as reference. The attack procedure is described in step by step. The comparison of the complexities is also given in section. Since it is essential to guess all 128 keys of initial round, this attack is not suitable for the key recovery of AES-128 or AES-192.

The complexity of an attack on any *n*-bit block cipher of key size *k* is distributed among three parameters: data, memory and time complexities. The complexity trade-off between these three parameters results in 3 basic attacks, called dictionary attack, codebook attack and exhaustive search respectively. Exhaustive search try all the possible keys (2^k) to attack the system. If an attack can retrieve the keys with a lesser time complexity than exhaustive search, it is considered as a successful attack. Time complexity is less expensive than the other two but considered sometimes as the most important parameter since time cannot be replaced by money.

This paper is organized as follows. Section 2 gives a brief description about AES operations. Section 3 tells about the notations used in this paper. Section 4 elaborates about the 4 round impossible differentials used in the conventional attacks. Section 5 details the procedure of first 8 round impossible differential cryptanalysis against AES - 256 in step by step. The data and time complexity calculation is also carried out here. Section 6 gives the features of one round lowered 8 round impossible differential attack against AES-256 and explains the procedure of cryptanalysis in step by step. Comparison of the complexities is also given. Section 7 concludes the topic with future scope

2. STRUCTURE OF AES

The data input to an AES system is of the size 128 bits, but the key size varies as 128, 192 and 256 bits for different variants. The representation of arrangement of 16 bytes of plain texts and its intermediate stages used in this paper is shown in Fig. 1. Here AES represents the variant with a key size of 256 bits

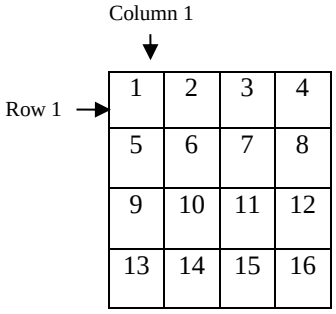

Figure 1. 4×4 Byte indexing of 128 bit data block

A. *Each round does the following 4 different operations.*

1) Substitution of Bytes (SB): The data is applied to a substitution box so that each byte gets substituted with another byte, defined by the SBox. The real data gets hidden and creates confusion.
2) Shift Rows (SR): Providing a cyclic shift to each row. Each bytes in the i^{th} row is shifted *i-1* times to left where *i* = 1, 2, 3, 4. This operation diffuses all the bytes except the members in the first row.
3) Mix Columns (MC): Mix Column operation redefines each member in a particular column. This is achieved by multiplying each column with a 4×4 constant matrix *M* over the field GF(2^8), where M is given by

$$M = \begin{pmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{pmatrix}$$

4) Add Round Key (AK): Entire 128 bits of data is encrypted with same number of key bits. Exclusive OR operation is performed here. Key is generated from the initial key applied as per the key generation schedule.

All the operations are reversible. A reverse substitution box is used for inversing the substitution operation. To get the inverse Shift Row operation, each byte in 2nd, 3rd and 4th rows are shifted 1, 2 and 3 positions respectively towards right. Inverse of Mix Column operation is obtained by multiplying each column with a 4×4 constant matrix M' over the field GF(2^8), where M' is given by

$$M' = \begin{pmatrix} e & b & d & 9 \\ 9 & e & b & d \\ d & 9 & e & b \\ b & d & 9 & e \end{pmatrix}$$

To get the Inverse of AK operation, the same secret key is Exclusively ORed with the 128 bit cipher text

3. NOTATIONS

In this paper the following Notations are used. Let the initial whitening sub key be K_0 and K_i be the i^{th} round key. The notations SB_i , SR_i and MC_i and AK_i are used to denote Substitution of Bytes, Shift Rows, Mix Columns and Add Round Key respectively, of i^{th} round. The notations SB_i^{-1} , SR_i^{-1}, and MC_i^{-1} are used to denote the inverse operations of Substitution of Bytes, Shift Rows, and Mix Columns respectively, of i^{th} round. For inversing the Add Key Round the same operation is done with the same key. So it doesn't need another symbol. In the second attack 'S' represents the initial input of a conventional impossible differential. '*C*' represents the intermediate data calculated backward from 1st round for adding one round before the first round of conventional impossible attack and '*I*' represents the input data.

4. FOUR ROUND IMPOSSIBLE DIFFERENTIAL

The impossible condition is used in differential cryptanalysis to eliminate the wrong keys. For AES, if same keys are used to derive differentials, maximum number of rounds involved in creating an impossible differential is four. The impossible condition is formed by using two differentials, one in forward direction and the other in reverse direction. For this, all input byte difference of the 1st round is made zero, called hereafter passive bytes, except in one, which is called an active byte. At the end of two rounds, it can be seen that all byte differences becomes active, on which no further analysis is possible. Suppose, the 4th round output has passive byte difference in any of the following byte positions: (1, 8, 11, 14), (2, 5, 12, 15), (3, 6, 9, 16), or (4, 7, 10, 13). Such a differential when decrypted two rounds will have 4 passive bytes differentials. This condition can't happen while decrypting AES with more than 4 rounds, if right keys are guessed. So, occurrence of an impossible condition points to a wrong key guess.

Consider two 16 byte plain texts which have equal values in all bytes except one. Now we have one active byte and 15 passive bytes. Let 'a' be the value of active byte. Here '*a'* denotes a known value, '*N'* denotes any non-zero value and '?' denotes any value. Fig. 2 shows the propagation of the differentials. The operation SB_1makes the value of active byte as 'N'. The same is retained after SR_1 , but MC_1 makes all members of 1stcolumn active. The operation SR_2 shifts these 4 active bytes to different columns and as a result, MC_2 makes all 16 bytes active. Since same keys are used to generate the Round 1: differentials, Add Round Key operations do not alter the differentials

It is assumed that the 4th round output has 4 passive bytes in positions 1, 8, 11 and 14. All other bytes difference is assumed as 'any value'. This assumption is to create an impossible condition. The reverse shift row operation SR_4^{-1} leaves all passive bytes in first column. It can be seen that AK_3 and MC_3^{-1} doesn't change the status of differentials. The operation SR_3^{-1} shift the rows in reverse order so that bytes 1, 6, 11and 16 becomes passive and SB_3^{-1} doesn't alter this status. Now the output of second round carries all active bytes but the input of third round has four passive bytes. This is a contradiction which can't happen with the guess of right key candidates.

5. FIRST ATTACK ON EIGHT ROUND AES

In this attack, the above mentioned four round impossible differentials is applied between round two and round five. Keys of some of the 8th round byte positions are assumed. To reduce the number of key byte assumptions, the order of operations Add Round Key and Mix Columns in the rounds 7, 6 and 5 are interchanged. In order to nullify the effect of this interchange, K_7, K_6 and K_5 are substituted by K_7^*, K_6^* and K_5^* respectively. The attack is illustrated in Fig. 3. The term 'Prob.' means that the state change occurs with a probability less than one

A. Attack Procedure

Precomputation: Calculate the 2^{32} values of bytes in positions 1, 6, 11, 16 which will lead the operation MC_1 to have a difference of 4 possible combinations (a,0,0,0), (0,a,0,0), (0,0,a,0), (0,0,0,a). Create a Hash table H_p and store these $2^{32}\times4\times(2^8\text{-}1) = 2^{42}$ pairs of 4 bytes values indexed by the Ex-OR differences. Here one indexed value corresponds to $2^{42}/2^{32} = 2^{10}$ values.

B. Algorithm:

1) Choose a set of 2^{32} plaintexts of 16 bytes in which all the bytes are fixed except bytes in positions 1, 6, 11, 16. This is called a structure. The number of possible pairs which can be formed out of a structure is $2^{32}\times (2^{32}-1)/2 = 2^{63}$. Select m number of different structures, which gives $2^{32}m$ plaintexts and $2^{63}m$ plaintext pairs.

$$\begin{pmatrix} a & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \xrightarrow{SB_1} \begin{pmatrix} N & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \xrightarrow{SR_1} \begin{pmatrix} N & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \xrightarrow{MC_1} \begin{pmatrix} N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \end{pmatrix} \xrightarrow{AK_1} \begin{pmatrix} N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \end{pmatrix}$$

Round 2:

$$\xrightarrow{SB_2} \begin{pmatrix} N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \end{pmatrix} \xrightarrow{SR_2} \begin{pmatrix} N & 0 & 0 & 0 \\ 0 & 0 & 0 & N \\ 0 & 0 & N & 0 \\ 0 & N & 0 & 0 \end{pmatrix} \xrightarrow{MC_2} \begin{pmatrix} N & N & N & N \\ N & N & N & N \\ N & N & N & N \\ N & N & N & N \end{pmatrix} \xrightarrow{AK_2} \begin{pmatrix} N & N & N & N \\ N & N & N & N \\ N & N & N & N \\ N & N & N & N \end{pmatrix}$$

Contradiction

Round 3:

$$SB_3^{-1} \begin{pmatrix} 0 & ? & ? & ? \\ ? & 0 & ? & ? \\ ? & ? & 0 & ? \\ ? & ? & ? & 0 \end{pmatrix} \xleftarrow{SR_3^{-1}} \begin{pmatrix} 0 & ? & ? & ? \\ 0 & ? & ? & ? \\ 0 & ? & ? & ? \\ 0 & ? & ? & ? \end{pmatrix} \xleftarrow{MC_3^{-1}} \begin{pmatrix} 0 & ? & ? & ? \\ 0 & ? & ? & ? \\ 0 & ? & ? & ? \\ 0 & ? & ? & ? \end{pmatrix} \xleftarrow{AK_3} \begin{pmatrix} 0 & ? & ? & ? \\ 0 & ? & ? & ? \\ 0 & ? & ? & ? \\ 0 & ? & ? & ? \end{pmatrix}$$

Round 4:

$$\xleftarrow{SB_4^{-1}} \begin{pmatrix} 0 & ? & ? & ? \\ 0 & ? & ? & ? \\ 0 & ? & ? & ? \\ 0 & ? & ? & ? \end{pmatrix} \xleftarrow{SR_4^{-1}} \begin{pmatrix} 0 & ? & ? & ? \\ ? & ? & ? & 0 \\ ? & ? & 0 & ? \\ ? & 0 & ? & ? \end{pmatrix} \xleftarrow{AK_4} \begin{pmatrix} 0 & ? & ? & ? \\ ? & ? & ? & 0 \\ ? & ? & 0 & ? \\ ? & 0 & ? & ? \end{pmatrix}$$

Figure 2. A 4 round impossible differential

Round 1:

$$\begin{pmatrix} N & 0 & 0 & 0 \\ 0 & N & 0 & 0 \\ 0 & 0 & N & 0 \\ 0 & 0 & 0 & N \end{pmatrix} \xrightarrow{SR_1} \begin{pmatrix} N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \end{pmatrix} \xrightarrow{MC_1} \begin{pmatrix} a & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \xrightarrow{AK_1} \begin{pmatrix} a & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

……4 Round Impossible Differential……

$$\begin{pmatrix} 0 & 0 & 0 & ? \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & ? \\ 0 & 0 & 0 & ? \end{pmatrix} \xleftarrow{MC_5^{-1}} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & N \\ 0 & 0 & 0 & N \end{pmatrix}$$ Prob.

$$\xleftarrow{SB_6^{-1}} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & N \\ 0 & 0 & 0 & N \end{pmatrix} \xleftarrow{SR_6^{-1}} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & N & 0 & 0 \\ N & 0 & 0 & 0 \end{pmatrix} \xleftarrow{AK_6} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & N & 0 & 0 \\ N & 0 & 0 & 0 \end{pmatrix} \xleftarrow{MC_6^{-1}} \begin{pmatrix} N & N & 0 & 0 \\ N & N & 0 & 0 \\ N & N & 0 & 0 \\ N & N & 0 & 0 \end{pmatrix}$$ Prob.

$$\xleftarrow{SB_7^{-1}} \begin{pmatrix} N & N & 0 & 0 \\ N & N & 0 & 0 \\ N & N & 0 & 0 \\ N & N & 0 & 0 \end{pmatrix} \xleftarrow{SR_7^{-1}} \begin{pmatrix} N & N & 0 & 0 \\ N & 0 & 0 & N \\ 0 & 0 & N & N \\ 0 & N & N & 0 \end{pmatrix} \xleftarrow{AK_7} \begin{pmatrix} N & N & 0 & 0 \\ N & 0 & 0 & N \\ 0 & 0 & N & N \\ 0 & N & N & 0 \end{pmatrix} \xleftarrow{MC_7^{-1}} \begin{pmatrix} N & N & N & 0 \\ N & N & 0 & N \\ N & 0 & N & N \\ 0 & N & N & N \end{pmatrix}$$ Prob.

$$\xleftarrow{SB_8^{-1}} \begin{pmatrix} N & N & N & 0 \\ N & N & 0 & N \\ N & 0 & N & N \\ 0 & N & N & N \end{pmatrix} \xleftarrow{SR_8^{-1}} \begin{pmatrix} N & N & N & 0 \\ N & 0 & N & N \\ N & N & N & 0 \\ N & 0 & N & N \end{pmatrix} \xleftarrow{AK_8} \begin{pmatrix} N & N & N & 0 \\ N & 0 & N & N \\ N & N & N & 0 \\ N & 0 & N & N \end{pmatrix}$$

Figure 3. 8 round attack towards AES - 256

2) Choose only the plaintext pairs whose cipher text output difference of 8th round is non zero in all byte positions except 4, 6, 12 and 14. Expected number of such plain text pairs is $2^{63}m \times 2^{-32} = 2^{31}m$.
3) Make a list A of possible 2^{32} values of the key bytes in positions 1, 6, 11, 16 of K_0
4) Perform the one round decryption operations after guessing the key values of byte positions 1, 8 and 11 of 8th round. Calculate the difference in ciphertext pairs after MC_7^{-1}. Accept the pair if the byte differences in positions 1and 5 are non zero and byte differences in remaining positions is zero. The probability for the existence of such a pair is 2^{-16}. The number of remaining pairs now is $2^{15}m$.
5) Perform the one round decryption operations after guessing the key values of byte positions 2, 5 and 15 of 8th round. Calculate the difference in cipher text pairs after MC_7^{-1}. Accept the pair if the byte differences in positions 2 and 14 are non zero and byte differences in remaining positions is zero. The probability for the existence of such a pair is 2^{-16}. The number of remaining pairs now is $2^{-1}m$.
6) Perform the one round decryption operations after guessing the key values of byte positions 3, 9 and 16 of 8th round. Calculate the difference in ciphertext pairs after MC_7^{-1}. Accept the pair if the byte differences in positions 11 and 15 are non zero and byte differences in remaining positions is zero. The probability for the existence of such a pair is 2^{-16}. The number of remaining pairs now is $2^{-17}m$.
7) Perform the one round decryption operations after guessing the key values of byte positions 7, 10 and 13 of 8th round. Calculate the difference in cipher text pairs after MC_7^{-1}. Accept the pair if the byte differences in positions 1, 2 are non zero and byte differences in remaining positions is zero. The probability for the existence of such a pair is 2^{-16}. The number of remaining pairs now is $2^{-33}m$.
8) Guess the key values of byte positions 1, 8, 11, 14 of 7th round and perform AK_7, SR_7^{-1}, SB_7^{-1} and MC_6^{-1}. Discard the pair if difference of byte pairs 1, 5, and 9 are not zero. Probability for getting such a pair is 2^{-24}. Now the number of remaining pairs is $2^{-57}m$.
9) Guess the key values of byte positions 2, 5, 12, 15 of 7th round and perform AK_7, SR_7^{-1}, SB_7^{-1} and MC_6^{-1}. Discard the pair if difference of byte pairs 2, 6, and 14 are not zero. Probability for getting such a pair is 2^{-24}. Now the number of remaining pairs is $2^{-81}m$.
10) Guess the key values of bytes 10 and 13 of 6th round and perform AK_6, SR_6^{-1}, SB_6^{-1} and MC_5^{-1}. Check if one of the four bytes is zero. If no, discard the pair. Probability for getting such a pair is 2^{-6} so that the number of remaining pairs is $2^{-87}m$.
11) For the remaining pairs, access the pairs P_1 and P_2 corresponding to the bytes difference in 1, 6, 11, 16. Calculate $P_d = P_1 \oplus P_2$. Find the values of x and y in hash table H_P corresponding to P_d. Eliminate the key value $P_1 \oplus x$ from table A.
12) Out the values of A, with the guessed key values 1, 2, 3, 5, 6, 7, 9, 10, 11, 13,14, 15 of 8th round, 1, 2, 5, 8, 11, 12, 14, 15 of 7th round and 10, 13 of 6th round, if A is not empty. So 26 key bytes are retrieved.

C. *Complexity Analysis*

1) *Data Complexity Analysis:*

Step 2 leaves $2^{31}m$ pairs remained. After Steps 4,5, 6 and 7 the available number of pairs becomes $2^{-33}m$. Step 8 again reduce the available number of pairs to $2^{-57}m$. Step 9 leaves $2^{-81}m$ pairs and finally remaining pairs becomes $2^{-81}m$ after step 10 for a guess of 12 bytes in 8th round, 8 bytes in 7th round and 2 bytes in 6th round. Therefore the number of subkeys that can be expected from table A for a given key guess is $2^{-32}(1-\frac{2^{-10}}{2^{-32}})^{m'}$. Only $2^{-32} \times e^{-2^{-6.5}} = 2^{-98.5}$ wrong values of 4 byte key will remain in A if $m' = 2^{29.5}$. Here $m = 2^{29.5} \times 2^{87} = 2^{116.5}$ structures, therefore the number total chosen plain text pairs required = $2^{116.5} \times 2^{32} = 2^{148.5}$.

TABLE I.TIME COMPLEXITY CALCULATION OF ECAH STEP

Head	operation	Number of operations	Value
Step 4	One Round Encryption	$2^{147.5} \times 2 \times 2^{24}/4$	$2^{170.5}$
Step 5	One Round Encryption	$2^{131.5} \times 2 \times 2^{48}/4$	$2^{178.5}$
Step 6	One Round Encryption	$2^{115.5} \times 2 \times 2^{72}/4$	$2^{186.5}$
Step 7	One Round Encryption	$2^{99.5} \times 2 \times 2^{96}/4$	$2^{194.5}$
Step 8	One Round Encryption	$2^{83.5} \times 2 \times 2^{128}/4$	$2^{210.5}$
Step 9	One Round Encryption	$2^{59.5} \times 2 \times 2^{160}/4$	$2^{218.5}$
Step 10	One Round Encryption	$2^{35.5} \times 2 \times 2^{176}/4$	$2^{210.5}$
Step 11	Memory Access to A	$2^{29.5} \times 2^{10} \times 2^{176}$	$2^{215.5}$

2) Time Complexity Analysis:

The attack time complexity of different steps mentioned in section 5 is given in Table I. The total time complexity of the attack can be calculated by adding individual time complexities required for each step. Total Time complexity ≈ $2^{215.5}$ 8 round AES encryptions. The attack proposed in this paper has the least time complexity for the least memory requirement.

6. ONE ROUND LOWERED IMPOSSIBLE DIFFERENTIAL ATTACK ON AES –256

Here, conventional 7 round attack given in [2] is modified to 8 round attack. Features of one round lowered impossible differential attack are given below. The attack is illustrated in Fig. 4

A. Features of one round lowered impossible differential attack

1) To reduce the attack complexity, the four round impossible differentials is applied between round 2 and round 5 instead of is applying it between round 1 and round 4 as in the conventional impossible differential attacks.
2) The input data pairs to 2^{nd} round (instead of round 1 in conventional) required to give non zero difference in byte positions 1, 6, 11, 16 and zero difference in all other byte positions are calculated
3) Input text pairs to round 0 are derived in the backward direction from the data calculated in step 2. This pre-computation reduces the total time complexity of attack. The run time complexity of decryption process is also reduced because the impossible condition now ends at 5^{th} round.
4) Since calculation of input text pairs to round 0 requires the whole key bytes of 0^{th} round, all the 128 key bits is to be assumed. So this concept of attack is not suitable of AES - 128.
5) Since only up to 7^{th} round attacks can be accomplished below the time complexity of 2^{128}, the proposed concept is suitable only for 8^{th} round attack or for higher rounds.
6) Since the plain text pairs required for the 0^{th} round is computed from 1^{st} round data, the data complexity required for *n* round attack is equal to that of *n-1* round conventional impossible attack.
7) In conventional 8^{th} round attack, Mix Column and Add Round Key operations of 7^{th} round are interchanged for reducing the number of subkey guess. Here it is not required so that the calculation for key replacement in 7^{th} round is not needed

B. Attack Procedure

Precomputation: Calculate the 2^{32} values of bytes in positions 1, 6, 11, 16 of 1^{st} round (denoted by 'S') which will lead the operation MC_2 to have a difference of 4 possible combinations (a,0,0,0), (0,a,0,0), (0,0,a,0), (0,0,0,a). Create a Hash table H_p and store these $2^{32}\times4\times(2^8\text{-}1) = 2^{42}$ pairs of 4 bytes values indexed by the Ex-OR differences. Here one indexed value corresponds to $2^{42}/2^{32} = 2^{10}$ values. Assume 128 bits keys of Round 0 and calculate the C values using one round decryption of the pairs. The memory required to store the C values will be 2^{42} bytes.

Round 0:

$$\begin{pmatrix} I & I & I & I \\ I & I & I & I \\ I & I & I & I \\ I & I & I & I \end{pmatrix} \xrightarrow{AK_0} \begin{pmatrix} C & C & C & C \\ C & C & C & C \\ C & C & C & C \\ C & C & C & C \end{pmatrix}$$

Round 1:

$$\xrightarrow{SB_1} \begin{pmatrix} C & C & C & C \\ C & C & C & C \\ C & C & C & C \\ C & C & C & C \end{pmatrix} \xrightarrow{SR_1} \begin{pmatrix} C & C & C & C \\ C & C & C & C \\ C & C & C & C \\ C & C & C & C \end{pmatrix} \xrightarrow{MC_1} \begin{pmatrix} S & 0 & 0 & 0 \\ 0 & S & 0 & 0 \\ 0 & 0 & S & 0 \\ 0 & 0 & 0 & S \end{pmatrix} \xrightarrow{AK_1} \begin{pmatrix} N & 0 & 0 & 0 \\ 0 & N & 0 & 0 \\ 0 & 0 & N & 0 \\ 0 & 0 & 0 & N \end{pmatrix}$$

Round 2:

$$\xrightarrow{SB_2} \begin{pmatrix} N & 0 & 0 & 0 \\ 0 & N & 0 & 0 \\ 0 & 0 & N & 0 \\ 0 & 0 & 0 & N \end{pmatrix} \xrightarrow{SR_2} \begin{pmatrix} N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \\ N & 0 & 0 & 0 \end{pmatrix} \xrightarrow{MC_2} \begin{pmatrix} a & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \xrightarrow{AK_2} \begin{pmatrix} a & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

……4 Round Impossible Differential……

$$\begin{pmatrix} 0 & 0 & 0 & ? \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & ? \\ 0 & 0 & 0 & ? \end{pmatrix} \xleftarrow{MC_6^{-1}} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & N \\ 0 & 0 & 0 & N \end{pmatrix}$$

$$\xleftarrow{SB_6^{-1}} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & N \\ 0 & 0 & 0 & N \end{pmatrix} \xleftarrow{SR_6^{-1}} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & N & 0 & 0 \\ N & 0 & 0 & 0 \end{pmatrix} \xleftarrow{AK_6} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & N & 0 & 0 \\ N & 0 & 0 & 0 \end{pmatrix} \xleftarrow{MC_6^{-1}} \begin{pmatrix} N & N & 0 & 0 \\ N & N & 0 & 0 \\ N & N & 0 & 0 \\ N & N & 0 & 0 \end{pmatrix}$$

$$\xleftarrow{SB_7^{-1}} \begin{pmatrix} N & N & 0 & 0 \\ N & N & 0 & 0 \\ N & N & 0 & 0 \\ N & N & 0 & 0 \end{pmatrix} \xleftarrow{SR_7^{-1}} \begin{pmatrix} N & N & 0 & 0 \\ N & 0 & 0 & N \\ 0 & 0 & N & N \\ 0 & N & N & 0 \end{pmatrix} \xleftarrow{AK_7} \begin{pmatrix} N & N & 0 & 0 \\ N & 0 & 0 & N \\ 0 & 0 & N & N \\ 0 & N & N & 0 \end{pmatrix}$$

Figure 4. One Round lowered Impossible Differential attack on AES

C. Algorithm:

1) Choose a set of 2^{32} plaintexts of 16 bytes in which all the bytes are fixed except bytes in positions 1, 6, 11, 16, denoted by 'S'. This is called a structure. The number of possible pairs which can be formed out of a structure is $2^{32} \times (2^{32} - 1)/2 = 2^{63}$. Select m number of different structures, which gives $2^{32}m$ plaintexts and $2^{63}m$ plaintext pairs.
2) Calculate the differentials of the plain text pairs 'C' through operations MC_1^{-1}, SR_1^{-1}, SB_1^{-1}. Guess 128 bytes of whitening key K_0 and find the calculated input values 'I's. This is done for adding an extra round to the conventional impossible attack, so that the 4 round impossible differentials is lowered to one round. These calculations will not increase the time complexity, since it can be done parallel to the coming steps. (Pre-calculation of 'I' values demands a huge memory ($2^{33}m$), so run time calculation is adopted here)
3) Choose only the plaintext pairs whose cipher text output difference of 7th round is zero in all byte positions except 10 and 13. Expected number of such plain text pairs is $2^{63}m \times 2^{-64} = 2^{-1}m$.
4) Guess the key values of byte positions 1, 8, 11, 14 of 7th round and perform AK_7, SR_7^{-1}, SB_7^{-1} and MC_6^{-1}. Discard the pair if difference of byte pairs 1, 5, and 9 are not zero. Probability for getting such a pair is 2^{-24}. Now the number of remaining pairs is $2^{-25}m$.

TABLE II.COMPARISON OF COMPLEXITIES – FIRST ATTACK

Reference	Attack Type	Cipher	Rounds	Chosen Plain Text	Time complexity (8 round encryptions)	Memory
[2]	Impossible Differential	AES - 256	8	$2^{116.5}$	$2^{247.5}$	2^{45}
This paper	Impossible Differential	AES - 256	8	$2^{148.5}$	$2^{215.5}$	2^{45}

TABLE III.COMPARISON OF COMPLEXITIES – SECOND ATTACK

Reference	Attack Type	Rounds	Chosen Plain Text	Time complexity (8 round encryptions)	Memory	No of key bytes recovered
[2]	Impossible Differential	8	$2^{116.5}$	$2^{247.5}$	2^{45}	30
This paper	Impossible Differential	8	$2^{115.5}$	2^{247}	2^{45}	30

5) Guess the key values of byte positions 2, 5, 12, 15 of 7th round and perform AK_7, SR_7^{-1}, SB_7^{-1} and MC_6^{-1}. Discard the pair if difference of byte pairs 2, 6, and 14 are not zero. Probability for getting such a pair is 2^{-24}. Now the number of remaining pairs is $2^{-49}m$.
6) Make a list A of possible 2^{32} values of the key bytes in positions 1, 6, 11, 16 of K_1(Instead of K_0 in conventional impossible differential attack)
7) Guess the key values of bytes 10 and 13 of 6th round and perform AK_6, SR_6^{-1}, SB_6^{-1} and MC_5^{-1}. Check if one of the four bytes is zero. If no, discard the pair. Probability for getting such a pair is 2^{-6} so that the number of remaining pairs is $2^{-55}m$.
8) For the remaining pairs, access the pairs P_1 and P_2 corresponding to the bytes difference in 1, 6, 11, 16. Calculate $P_d = P_1 \oplus P_2$. Find the values of x and y in hash table H_P corresponding to P_d. Eliminate the key value $P_1 \oplus x$ from table A.
9) Out the values of A, if A is not empty, with the guessed key values of all bytes of 0th round, 1, 2, 5, 8, 11, 12, 14, 15 of 8th round and 10, 13 of 7th round, i.e, 30 key bytes.

D. Complexity Calculations

Step 2 leaves $2^{-1}m$ pairs remained. Step 3 again reduce the available number of pairs to $2^{-25}m$. Step 4 leaves $2^{-49}m$ pairs and finally remaining pairs becomes $2^{-55}m$ after step 5 for a guess of 8 bytes in 8th round, 2 bytes in 7th round. Therefore the number of sub keys that can be expected from table A for a given key guess is $2^{-32}(1-\frac{2^{-10}}{2^{-32}})^{m'}$. Only $2^{-32} \times e^{-2^{-6.5}} = 2^{-98.5}$ wrong values of 4 byte key will remain in A if $m' = 2^{28.5}$. Here $m = 2^{28.5} \times 2^{55} = 2^{83.5}$ structures, therefore the number total chosen plain text pairs required = $2^{83.5} \times 2^{32} = 2^{115.5}$.

Time complexity of the 7 round attack [2] is given as 2^{119}. Since the round zero guess 2^{128} key bytes, the overall time complexity of the attack is given by $2^{119} \times 2^{128} = 2^{247}$ 8 round AES encryptions

Comparison of complexities for the first and second attack mentioned in this paper is given in Table II and Table III respectively.

7. CONCLUSION

Two 8 round impossible differential cryptanalyses are proposed in this paper against AES-256. These attacks show how an existing attack can be modified according to the requirements. The first attack reduces the time complexity to a significant extent but at the cost of data complexity and with no change in memory requirement. The second attack reduces both time and data complexity by lowering the rounds with which the impossible condition is created. Comparison of the complexities for both attacks is also given. Based on the techniques given in this paper, other existing differential attacks also can be modified based on the requirements.

REFERENCES

[1] Nat'l Institute of Standards and Technologies "Announcing the Advanced Encryption Standard (AES)," Fed. Information Processing Standards Publication, no. 197, Nov. 2001

[2] Wentao Zhang, Wenling Wu, Dengguo Feng. "New Results on Impossible Differential Cryptanalysis of Reduced AES". Lecture Notes in Computer Science 4817, pp- 239-250, Springer - Verlag Berlin Heidelberg – 2007

[3] Biham, E., Biryukov, A., Shamir, A. "Cryptanalysis of Skipjack Reduced to 31 Rounds" Stern, J (ed.) EUROCRYPT 1999. LNCS, vol.1592, pp 12-23. Springer, Heidelberg, 1999.

[4] Biham, E., Keller, N."Cryptanalysis of Reduced Variants of Rijndael, in Official Public comment for Round 2 of the AES development effort (2000)," *available at http://csrc.nist.gov/encryption/aes/round2/conf3/aes3papers.html*

[5] Cheon J.H., Kim M., Kim K., Lee J., -Y., Kang S., "Improved Impossible Differential Cryptanalysis of Rijndael and Crypton," *ICISC 2001 LNCS*, vol. 2288, pp. 39-49, Springer, Heidelberg, 2002

[6] Phan R.C. –W: "Impossible differential Cryptanalysis of Seven Round Advanced Encryption Standard (AES)," Information Processing Letters 91(1) pp. 33-38, 2004

[7] A.Biryukov,O.Dunkelman,N.Keller,D.Khovratovich,andA.Shamir , "Key recovery attacks of practical complexity on AES-256 variants with up to 10 rounds," inAdvances in cryptology—EUROCRYPT 2010, vol. 6110 ofLecture Notes in Comput. Sci.,pp.299–318,Springer,Berlin,2010.

[8] J. Cui, L. Huang, H. Zhong, and W. Yang, "Improved related-keyattack on 7-round AES-128/256," inProceedings of the ICCSE2010 - 5th International Conference on Computer Science andEducation, pp. 462–466, 2010.

[9] Bogdanov A., Khovratovich D., Rechberger C. (2011) "Biclique Cryptanalysis of the Full AES". In: Lee D.H., Wang X. (eds) Advances in Cryptology – ASIACRYPT 2011. ASIACRYPT 2011. Lecture Notes in Computer Science, vol 7073. pp. 344-371Springer, Berlin, Heidelberg

[10] Bogdanov A., Chang D., Ghosh M., Sanadhya S.K. (2015) "Bicliques with Minimal Data and Time Complexity for AES". In: Lee J., Kim J. (eds) Information Security and Cryptology - ICISC 2014. ICISC 2014. Lecture Notes in Computer Science, vol 8949. pp 160-174 Springer, Cham

[11] X.Zhao,S.Guo,F.Zhangetal.,"Acomprehensivestudyofmultiple deductions-based algebraic trace driven cache attackson AES,"Computers & Security,vol.39,pp.173–189,2013

[12] J.Kang,K.Jeong,J.Sung, S.Hong andK.Lee,"Collisionattacks on AES-192/256, Crypton-192/256, mCrypton-96/128,and anubis,"Journal of Applied Mathematics,vol.2013,ArticleID 713673, 10 pages, 2013

[13] S. Sahmoud, "Enhancement the Security of AES Against Modern Attacks by Using Variable Key Block Cipher,"Int. Arab J.e-Technol,vol.3,pp.17–26,2013

[14] D. Mukhopadhyay, "An improved fault based attack of theadvanced encryption standard,"Lect. Notes Comput. Sci.(including Subser. Lect.Notes Artif. Intell. Lect. Notes Bioinfor-matics),vol.5580,pp.421–434,2009

[15] C. H. Kim, "Differential faultanalysis against AES-192 and AES-256 with minimal faults," inProceedings of the 7th InternationalWorkshop on Fault Diagnosis and Tolerance in Cryptography,FDTC 2010,pp.3–9,USA,August2010

[16] M. Tunstall, D. Mukhopadhyay, and S. Ali, "Differential faultanalysis of the advanced encryption standard using a singlefault,"Inf. Secur. Theory Pract. Secur. Priv. Mob. Devices Wirel.Commun, pp. 224–233, 2011

[17] A.Barenghi,G.M.Bertoni,L.Breveglieri,andG.Pelosi,"Afault induction technique based on voltage underfeeding withapplication to attacks against AES and RSA,"The Journal ofSystems and Software,vol.86,no.7,pp.1864–1878,2013.

[18] H. Mestiri, F. Kahri, B. Bouallegue, and M. Mach-hout, "A high-speed aes design resistant to fault in-jection attacks,"Microprocessors and Microsystems,vol. 41, no. 1, pp. 47–55, 2016

[19] N. Farhady Ghalaty, B. Yuce, and P. Schaumont, "AnalyzingtheEfficiency of Biased-Fault Based Attacks,"IEEE EmbeddedSystems Letters,vol.8,no.2,pp.33–36,2016

[20] Mohamed Saied Emam Mohamed, Stanislav Buly-gin, Michael Zohner, Annelie Heuser, and MichaelWalter, "Improved algebraic side-channel attack onAES,"Journal of Cryptographic Engineering, vol. 3,no. 3, pp. 139–156, 2014.

[21] S. Patranabis, A. Chakraborty, D. Mukhopadhyay, and P. P.Chakrabarti, "Fault Space Transformation: A Generic Approachto Counter Differential Fault Analysis and Differential FaultIntensity Analysis on AES-Like Block Ciphers,"IEEE Transac-tions on Information Forensics and Security,vol.12,no.5,pp.1092–1102, 2017.

[22] Jithendra.K.B, Shahana.T.K, "New Results in Related Key Impossible Differential Cryptanalysis on Reduced Round AES-192", International Conference On Advances in Communication and Computing Technology (ICACCT), Sangamner, India, pp 291-295,IEEE Xplore -2018 February

[23] Mouha N., Wang Q., Gu D., Preneel B. (2012) "Differential and Linear Cryptanalysis Using Mixed-Integer Linear Programming". In: Wu CK., Yung M., Lin D. (eds) Information Security and Cryptology. Inscrypt 2011. Lecture Notes in Computer Science, vol 7537. Springer, Berlin, Heidelberg pp 57-76

[24] Zhu B., Dong X., Yu H. (2019) "MILP-Based Differential Attack on Round-Reduced GIFT". In: Matsui M. (eds) Topics in Cryptology – CT-RSA 2019.. Lecture Notes in Computer Science, pp. 372-390, vol 11405. Springer, Cham . 2019

Jithendra.K.B Completed B.Tech in Electronics and Communication from Rajiv Gandhi Institute of Technology, Kottayam, INDIA and M.Tech in Embedded Systems from National Institute of Electronics and Information Technology, Kozhikode, INDIA. Presently Working as Asst. Professor in College of Engineering, Thalassery, India and doing research from Cochin University of Science and Technology, Cochin. Areas of interests includes VLSI Design, ASIC Design, Digital System Design and Cryptography.

Shahana.T.K: Professor at Cochin University of Science and Technology (CUSAT), Kerala, India. She received her Ph. D. in VLSI Design from Cochin University of Science and Technology in 2009, M.Tech in Digital Electronics from CUSAT in 1999 and B.Tech in Electronics and Communication Engineering from Mahatma Gandhi University in 1997. Her research interests include VLSI implementation of digital systems, Digital Filters, Multi-standard wireless transceivers, RNS-based arithmetic circuits, Low-power design, Cryptography etc.