01409 - Lightweight Cryptography E15

Student name and id: Mikkel Ole Rømer (s113408)

Exercise Sheet Week 10

1 Exercise 1

Prove that the difference between the correct and faulty state at the end of round 8 is of form (1)

$$\Delta_r = \begin{bmatrix} \Delta_1 & 0 & 0 & 0 \\ \Delta_2 & 0 & 0 & 0 \\ \Delta_3 & 0 & 0 & 0 \\ \Delta_4 & 0 & 0 & 0 \end{bmatrix} \tag{1}$$

From (1) it is clear that the proposed difference is only expressed in row 1. Since this is the result of injecting one fault byte at position 0, in the beginning of round 8, and (1) is the difference at the end of round 8, it makes sense to look at the matrix transformation doing one single round.

Input Matrix

$$X = \begin{bmatrix} x_0 & x_4 & x_8 & x_{12} \\ x_1 & x_5 & x_9 & x_{13} \\ x_2 & x_6 & x_{10} & x_{14} \\ x_3 & x_7 & x_{11} & x_{15} \end{bmatrix} \tag{2}$$

Substitution byte operation

$$Y = \begin{bmatrix} y_0 & y_4 & y_8 & y_{12} \\ y_1 & y_5 & y_9 & y_{13} \\ y_2 & y_6 & y_{10} & y_{14} \\ y_3 & y_7 & y_{11} & y_{15} \end{bmatrix} \tag{3}$$

Shiftrow operation

$$Y = \begin{bmatrix} y_0 & y_4 & y_8 & y_{12} \\ y_5 & y_9 & y_{13} & y_1 \\ y_{10} & y_{14} & y_2 & y_6 \\ y_{15} & y_3 & y_7 & y_{11} \end{bmatrix} \tag{4}$$

Mixcolumn operation

It is clear so far (4), that the faulty byte at position 0, has not yet affected any other byte in the AES state. Only one operation could alter this scenario. The operation in mind is MixColumns. From (5) one operation of this procedure is shown on a single column vector. That is, four operations like (5) is performed through this procedure, one for each column in Y.

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 2a_0 + 3a_1 + 1a_2 + 1a_3 \\ 1a_0 + 2a_1 + 3a_2 + 1a_3 \\ 1a_0 + 1a_1 + 2a_2 + 3a_3 \\ 3a_0 + 1a_1 + 1a_2 + 2a_3 \end{bmatrix} \tag{5}$$

It is readily seen that the procedure of mixcolumns, only affects the column in each operation independently. Moreover, each column position in the result, $[b_0, b_1, b_2, b_3]$, is created through addition of all column positions of the input column $[y_0, y_1, y_2, y_3]$. (5).

This means, that after the mixcolumn operation, only the values of the first row will be affected by the injected fault. All other position remains intact. Lastly, note that one operation is not spoken upon. The very last operation of an AES round is the key addition. However, this operation consists only of XOR operation of the round key on the AES state. Thus the difference will remain only in the first column (1).

□

1.1 Exercise 2

Prove that the difference between the correct and faulty state at the end of round 9 is of the form (6)

$$\Delta_{r+1} = \begin{bmatrix} 2\Delta_1 & \Delta_4 & \Delta_3 & 3\Delta_2 \\ \Delta_1 & \Delta_4 & 3\Delta_3 & 2\Delta_2 \\ \Delta_1 & 3\Delta_4 & 2\Delta_3 & \Delta_2 \\ 3\Delta_1 & 2\Delta_4 & \Delta_3 & \Delta_2 \end{bmatrix} \tag{6}$$

From (6), the Δ output matrix is seen at the end of the 9Th round. In the following exercise this structure will be investigated.

Input Matrix

$$\Delta_r = \begin{bmatrix} \Delta_1 & 0 & 0 & 0 \\ \Delta_2 & 0 & 0 & 0 \\ \Delta_3 & 0 & 0 & 0 \\ \Delta_4 & 0 & 0 & 0 \end{bmatrix} \tag{7}$$

Substitution byte operation

$$Y = \begin{bmatrix} \Delta_1 & 0 & 0 & 0 \\ \Delta_2 & 0 & 0 & 0 \\ \Delta_3 & 0 & 0 & 0 \\ \Delta_4 & 0 & 0 & 0 \end{bmatrix} \tag{8}$$

The substitution byte operation, does change the difference in terms of value. However, the position in which the difference occurs remains intact.

Shiftrow operation

$$Y = \begin{bmatrix} \Delta_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Delta_2 \\ 0 & 0 & \Delta_3 & 0 \\ 0 & \Delta_4 & 0 & 0 \end{bmatrix} \tag{9}$$

From (9), it is seen how the deltas are shifted to each column. Thus it is realised that the injected difference of position 0, in the first round only affected column 0. However, in round $r + 1$ the entire state is affected. Moreover, by introducing the mixcolumn procedure, the state will be even more corrupted.

Mixcolumn operation

Column 0

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} \Delta_1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2\Delta_1 + 0 + 0 + 0 \\ 1\Delta_1 + 0 + 0 + 0 \\ 1\Delta_1 + 0 + 0 + 0 \\ 3\Delta_1 + 0 + 0 + 0 \end{bmatrix} \tag{10}$$

Column 1

$$\begin{bmatrix} b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \Delta_4 \end{bmatrix} = \begin{bmatrix} 0 + 0 + 0 + 1\Delta_4 \\ 0 + 0 + 0 + 1\Delta_4 \\ 0 + 0 + 0 + 3\Delta_4 \\ 0 + 0 + 0 + 2\Delta_4 \end{bmatrix} \tag{11}$$

Column 2

$$\begin{bmatrix} b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \Delta_3 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 + 0 + 1\Delta_3 + 0 \\ 0 + 0 + 3\Delta_3 + 0 \\ 0 + 0 + 2\Delta_3 + 0 \\ 0 + 0 + 1\Delta_3 + 0 \end{bmatrix} \tag{12}$$

Column 3

$$\begin{bmatrix} b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ \Delta_2 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 + 3\Delta_2 + 0 + 0 \\ 0 + 2\Delta_2 + 0 + 0 \\ 0 + 1\Delta_2 + 0 + 0 \\ 0 + 1\Delta_2 + 0 + 0 \end{bmatrix} \tag{13}$$

When combining all the columns (14) is obtained, which is equivalent to (6).

$$\Delta_{r+1} = \begin{bmatrix} 2\Delta_1 & 1\Delta_4 & 1\Delta_3 & 3\Delta_2 \\ 1\Delta_1 & 1\Delta_4 & 3\Delta_3 & 2\Delta_2 \\ 1\Delta_1 & 3\Delta_4 & 2\Delta_3 & 1\Delta_2 \\ 3\Delta_1 & 2\Delta_4 & 1\Delta_3 & 1\Delta_2 \end{bmatrix} \tag{14}$$

□

Exercise 3

Hence or otherwise devise a strategy to find the 10Th round key.

In the above exercises, the paper explained a space-synchronised fault injection in the beginning of the 8Th round. Moreover, it was illustrated how this fault injection affected the AES state at the end of round 8 and 9. It should be realised that this method, does not include knowing the original plain text nor the key. Thus making this a cipher-text only attack, which is the most powerful attack in the deck.

From the dependency, or delta, matrix of (14), it is possible to define a strategy, that could recover the 10Th round-key, and thus invert the entire key-schedule. This is done by clever guessing. That is, four each column guess the four bytes, such that the delta-column is satisfied. This deduction, requires guesses of 4 bytes in 8 columns. Thus the key space becomes 2^{32} for each column, dramatically reducing the security of the AES-128 cryptography-system. However, there are 4 columns, hence the entire key-space becomes $4 \times 2^{32} = 2^{34}$. But since we know the expected deltas in each column, this DFA search could be concluded in parallel[1], thus reducing the problem back to the 2^{32} complexity.

□

2 Refferences

1 - Yang Li, Shigeto Gomisawa, Kazuo Sakiyama, Kazuo Ohta - An Information Theoretic Perspective on the Differential Fault Analysis against AES - The University of Electro-Communications 1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585, Japan - https://eprint.iacr.org/2010/032.pdf