Assignment 3, Question No. 2, Differential Fault Attack on AES

Group 2:

– Correct Ciphertext1: 0xb21eeb73953e7a2771db222ecbeea788

– Correct Ciphertext2: 0xcabe3e9988d6666a96a39e7b659ca91

– Faulty Ciphertext1: 0x33cf60141fd2f121ff9a6126fefd03e6

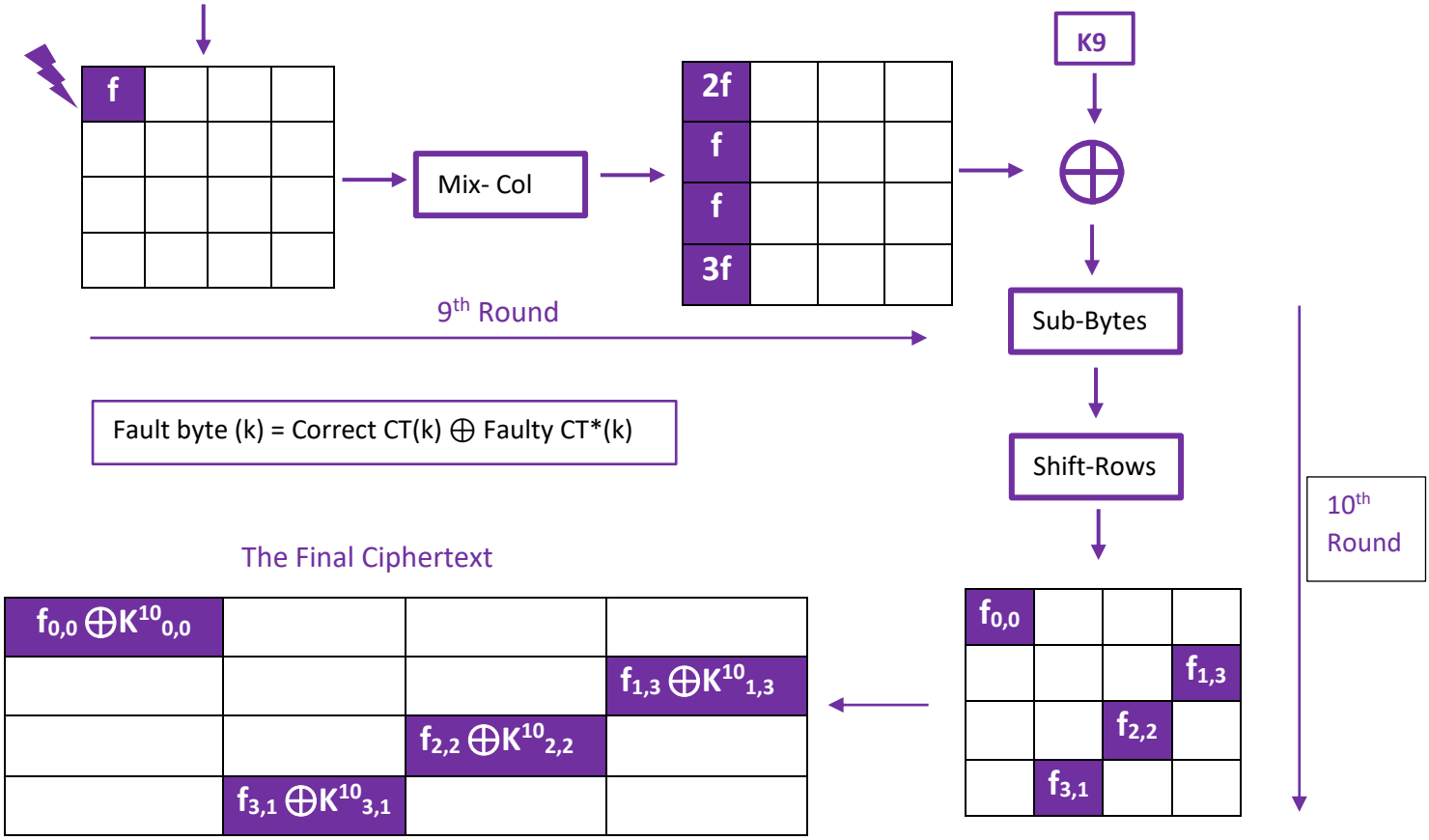
– Faulty Ciphertext2: 0x92317b8a9e24f1960f37385a4085b0d5

Problem and approach

We are provided with a pair of correct and faulty ciphertext, we are expected to formulate equations to recover one column of the key of round 10. This is the standard fault injection, injected in the start of the 9^{th} round in an AES, injected in the 0^{th} row and 0^{th} column Byte.

This fault is then expected to spread across an entire column after one round of the AES operation, with reverse engineering and publicly available algorithms of S-Box, we can formulate *inverse S-Box*, and we'll do the maths to reverse the *add round key* (using all possible guessed key for a Byte (i.e. 2^8 combinations)) we can do *shift rows* and *Mix-column* (using the irreducible polynomial theorem of GF(2^8) operation) to formulate 4 sets of equation.

The flow of equation generation/ Fault injection is as follows:



The formulated equations are:

Eqn1. $2f = S^{-1}(C_{0,0} \oplus K^{10}_{0,0}) \oplus S^{-1}(C^{*}_{0,0} \oplus K^{10}_{0,0})$

Eqn2. $f = S^{-1}(C_{1,3} \oplus K^{10}_{1,3}) \oplus S^{-1}(C^{*}_{1,3} \oplus K^{10}_{1,3})$

Eqn3. $f = S^{-1}(C_{2,2} \oplus K^{10}_{2,2}) \oplus S^{-1}(C^{*}_{2,2} \oplus K^{10}_{2,2})$

Eqn4. $3f = S^{-1}(C_{3,1} \oplus K^{10}_{3,1}) \oplus S^{-1}(C^{*}_{3,1} \oplus K^{10}_{3,1})$

So, we have created a python code who takes the two pairs of (C,C*) as input.

We then enumerate all possible keys for a Byte (i.e. for 8 bits, we get 2^8 possible keys, i.e. 256)

We then compute XOR of the (C and potential keys) and XOR of the (C* and potential keys)

We then compute inverse S-Box of the above two entities and compute the XOR of them.

We then form 6 set of equations from these 4 equations

(Eqn1=2*Eqn2),(Eqn1=2*Eqn3),(3*Eqn1=2*Eqn4),(2*Eqn2=Eqn3), (2*Eqn2=Eqn4), (3*Eqn3=Eqn4)

Here (*) is not multiplication, but is multiplication with irreducible polynomial of (GF(2^8): $X^8+X^4+X^3+X+1$)

Using these 6 set of equations, after round 1 (1st pair of C,C*) the potential keys, reduce from 256 to 30.

After the second round, we are left with our unique key column.

With given fault in the start of 9th round, the two pairs of CT and CT* and with the help of the 4 equations, we've successfully retrieved 4*(1B) of 10th round key (i.e. one column of key)

The extracted location of the key is ($K^{10}_{0,0}$, $K^{10}_{1,3}$, $K^{10}_{2,2}$, $K^{10}_{3,1}$)

We have followed a column major order throughout the code.

OUTPUT of our program:

```
C:\Users\souvik\OneDrive\Desktop> python q2.py

*******************************************************************************
 *****  Evaluation of step wise key-space reduction of target Bytes  *****
 *****  No. of potential keys in respective positions  *****
*******************************************************************************

Initially 2^8 keys possible for every Byte: k00: 256 |*|  k13: 256 |*| k22: 256 |*| k31: 256

*******************************************************************************

After Eqn 1: k00: 126  |*|  k13: 128  |*| k22: 256 |*| k31: 256

*******************************************************************************

After Eqn 2: k00: 62  |*|  k13: 128  |*| k22: 62  |*| k31: 256

*******************************************************************************
```

```
After Eqn 3: k00: 30  |*|  k13: 128  |*| k22: 62  |*| k31: 30

********************************************************************************

After Eqn 4: k00: 30  |*|  k13: 64  |*| k22: 62  |*| k31: 30

********************************************************************************

After Eqn 5: k00: 30  |*|  k13: 30  |*| k22: 62  |*| k31: 30

********************************************************************************

After Eqn 6: k00: 30  |*|  k13: 30  |*| k22: 30  |*| k31: 30

********************************************************************************

 *****  Second Iteration, with Correct & Faulty Ciphertext number 2 *****

********************************************************************************

After Eqn 7: k00: 3  |*|  k13: 3  |*| k22: 30  |*| k31: 30

********************************************************************************

After Eqn 8: k00: 1  |*|  k13: 3  |*| k22: 1  |*| k31: 30

********************************************************************************

After Eqn 9: k00: 1  |*|  k13: 3  |*| k22: 1  |*| k31: 1

********************************************************************************

After Eqn 10: k00: 1  |*|  k13: 1  |*| k22: 1  |*| k31: 1   Key Successfully Recovered

********************************************************************************

After Eqn 11: k00: 1  |*|  k13: 1  |*| k22: 1  |*| k31: 1

********************************************************************************

After Eqn 12: k00: 1  |*|  k13: 1  |*| k22: 1  |*| k31: 1

********************************************************************************

Final value of Bytes column 1 of key10 (in Hexadecimal) r0c0: 6d r1c3 83 r2c2 a3 r3c1 59

********************************************************************************
Final value of Bytes column 1 of key10 (in Binary) r0c0: [(0, 1, 1, 0, 1, 1, 0, 1)] r1c3 [(1,
0, 0, 0, 0, 0, 1, 1)] r2c2 [(1, 0, 1, 0, 0, 0, 1, 1)] r3c1 [(0, 1, 0,
1, 1, 0, 0, 1)]

********************************************************************************

PS C:\Users\souvik\OneDrive\Desktop>
```