# CSE653 - Topics in Cryptanalysis
# Mid Semester Examination - Winter 2024/2025

Name -
Roll No -

$22^{nd}$ February 2025
Total Marks - 20

---

## Section A
## Answer all questions

1. Define a linear function.
   Which of the following operation(s) is a non-linear operation? [1+1]

   - *addition modulo* $2^n$,
   - *circular rotation (left/right)*, and
   - *XOR*.

   - A function $f : X \to Y$, is a linear function iff $f(a + b) = f(a) + f(b)$, where $a, b \in X$ and the operation $+$ is defined on $f$. [1]

   - Addition modulo $2^n$ is a nonlinear function. [1]

2. Write **4** *attack models*, which are classified by the information that attackers can obtain, and are used to measure the security of a block cipher. [1]

   - Any of the four: Ciphertext only attack, known plaintext attack, chosen plaintext attack, chosen ciphertext attack, *adaptive versions*

3. How many differential characteristics with probability $2^{-55}$ exist for AES three rounds? [2]

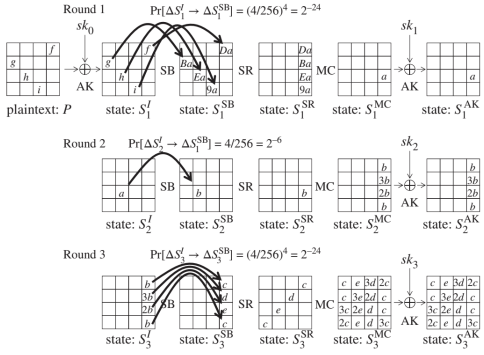



**Figure 4.14** Differential characteristic for AES three rounds with probability $2^{-54}$

Now we know the following two properties of AES S-box: For any input difference $\Delta(In)$:

- There is only one output difference with entry 4, i.e. $\Delta(In) \xrightarrow{2^{-6}} \Delta(Out)$
- There are 126 output differences with entry 2, i.e. $\Delta(In) \xrightarrow{2^{-7}} \Delta(Out)$

In the figure above, the characteristic is of probability $2^{-54}$, because we have considered only those S-box input-output pair with probability $2^{-6}$. Now if we want a characteristics with probability $2^{-55}$, we have change only one of the S-box input-output pair with that of probability $2^{-7}$. There are 126 such pairs for each input difference.

Since there are 9 such pairs which can be replaced, so the number of possible characteristics with probability $2^{-25}$ is $9 \times 126 = 1134$.

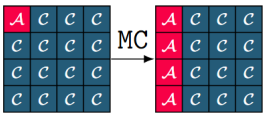
You do not need to draw the above figure in your answer. Just the computation as above will work. Just writing the answer will not be given any marks.

Any one who reached 126 as answer as above, i.e. forgot to multiply with 9 will also be given full marks.

## Section B
## Answer any 3 questions

4. Answer the following questions for integral attack on AES.

   (a) The propogation of the *All* property through the MixColumn operation is described as below. Explain why this propogation is correct. [3]




- We begin with a state as below, where only the first byte is active, and the others are constants:

$$\mathbf{c} = \begin{bmatrix} X \\ C \\ C \\ C \end{bmatrix}$$

where:
  - $X$ represents the active byte, and it can take all 256 possible values (i.e., $X \in [0, 255]$),
  - $C$ is a constant byte, which remains fixed.

In the MixColumns operation, each byte in a column is transformed by multiplying the byte by a constant matrix:

$$\mathbf{M} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

Applying the MixColumns operation on the column $\mathbf{c}$, the transformed column $\mathbf{c}'$ becomes:

$$\mathbf{c}' = \begin{bmatrix} (02 \cdot X) \oplus (03 \cdot C) \oplus (01 \cdot C) \oplus (01 \cdot C) \\ (01 \cdot X) \oplus (02 \cdot C) \oplus (03 \cdot C) \oplus (01 \cdot C) \\ (01 \cdot X) \oplus (01 \cdot C) \oplus (02 \cdot C) \oplus (03 \cdot C) \\ (03 \cdot X) \oplus (01 \cdot C) \oplus (01 \cdot C) \oplus (02 \cdot C) \end{bmatrix}$$

where:

- $\cdot$ represents multiplication in the finite field GF($2^8$),
- $\oplus$ represents the XOR operation.

## Breaking Down the MixColumns Output

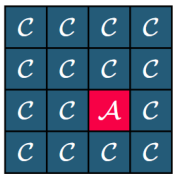
Let's break it down for each byte in $\mathbf{c}'$:

- **First byte** $c'_0$:
$$(02 \cdot X) \oplus (03 \cdot C) \oplus (01 \cdot C) \oplus (01 \cdot C)$$
This is the result of the first row of the matrix multiplied with the active byte and constants.
- **Second byte** $c'_1$:
$$(01 \cdot X) \oplus (02 \cdot C) \oplus (03 \cdot C) \oplus (01 \cdot C)$$
This is the result of the second row of the matrix.
- **Third byte** $c'_2$:
$$(01 \cdot X) \oplus (01 \cdot C) \oplus (02 \cdot C) \oplus (03 \cdot C)$$
This is the result of the third row of the matrix.
- **Fourth byte** $c'_3$:
$$(03 \cdot X) \oplus (01 \cdot C) \oplus (01 \cdot C) \oplus (02 \cdot C)$$
This is the result of the fourth row of the matrix.

Since $X$ takes all 256 possible values, the resulting bytes $c'_0, c'_1, c'_2, c'_3$ will each also take all 256 possible values.
Thus, even though we start with only one active byte $X$, after the MixColumns operation, all four bytes in the column become active.

Therefore, this propogation is correct.
First computing the matrix c' as above will have 1 marks. Computing the second part will have 2 marks

(b) Draw the integral property propogation in 3-round AES if the $10^{th}$ byte of the AES state in the first round has *All* property and other bytes have the *Constant property*, as shown in the figure below. [2]




You all know this answer. After the $3^{rd}$ we have all balanced bytes.

5. Consider that I have a block cipher with block size and key size = **64** bits. I want to recover the key using the TMTO attack. Consider that I have already computed the pre-computation table $\mathcal{T}$, for some plaintext $P$. Also assume you have the corresponding ciphertext $C$. Then answer the following questions:

(a) Write the psuedocode for recovering the key from the table [4]
Let $m$ be the number of the rows in the table $\mathcal{T}$ and $t$ be the value such that $m \times t = 2^{64}$. Then the algorithm for the key recovery is as follows:

**Algorithm 1:** Search for $K$ within the pre-computed chains

**Input:** Pre-computed chains and input $P$
**Output:** Key $K$ or a message indicating key not found

```
Function SearchKey:
    Y = E(C, P) ;
    for i = 0 to t − 1 do
        for j = 0 to m − 1 do
            if Y == EP_j then
                Find K in the chain beginning with SP_j;
                return K;
        Y = E(Y, P);
    return key not found;
```

(b) What is the **time complexity** of recovering the key of the given block cipher. [1]

- The time complexity is $T = 2^t$. The answer will depend on what you take $m$ and $t$. You have to give some values to $t$ and $m$, just writing $t$ as answer will have only 1/2 marks.

6. Consider that you have obtained a differential trail, with probability $2^{98}$, for 3-round AES. Then,

(a) Write a pseudocode for distinguishing a 3-round AES from a random permutation [4]

**Algorithm 2:** Distinguishing Attack against AES Reduced to 3 Rounds

**Input:** A differential characteristic propagating from $\Delta P$ to $\Delta C$ with probability $2^{-54}$.
**Output:** A determining bit $B \in \{0, 1\}$.

```
Choose 2^98 distinct plaintexts P_i for i = 1, 2, ..., 2^98;
for i ← 1 to 2^98 do
    Query P_i to the encryption oracle and obtain the corresponding ciphertext C_i;
    Query P'_i = P_i ⊕ ΔP to the encryption oracle and obtain the corresponding ciphertext C'_i;
    if C_i ⊕ C'_i = ΔC then
        return 0; // The oracle is AES reduced to 3 rounds.
return 1; // The oracle is a random permutation.
```

You can use any other notation for $\Delta P$ and $\Delta C$

(b) What is the **complexity** of this distinguishing attack. [1]
The complexity of this attack is $O(2^{98})$. $O(2^{99})$ will also be correct.

7. Consider that I swap the MixColumn and AddRoundKey operations in 1 round AES, as shown in the figure below. Answer the following questions:

(a) Will the ciphertexts $C$ and $C_1$ be the same or different for the same plaintext $P$? Consider that both the keys are all 1's. [1]

Yes. (The reason is that in the question, the keys are all assumed to be the same)

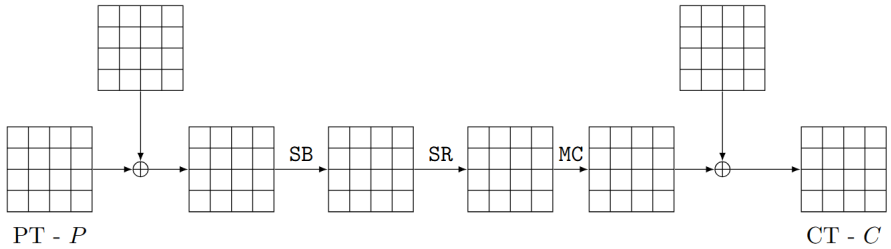
(b) If your answer is **Yes**, then explain why? If your answer is **No**, then explain why? [4]
Since the input $P$ is same in both, the output after the shift rows operation is same in both the figures. Let the first column after SR be $[a, b, c, d]$. Now let us compute the first byte of the ciphertext in both the cases one by one:

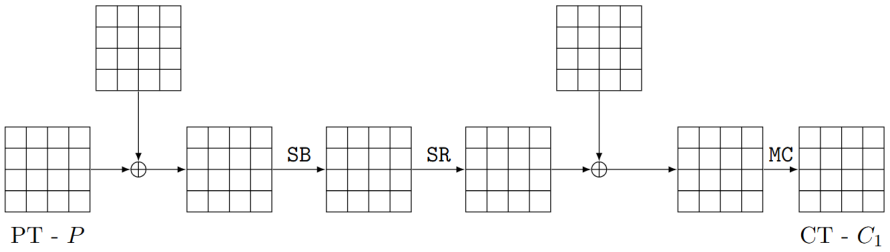
- **Case 1**: In the first case, the first byte after MC will be:

$$(02 \cdot a) \oplus (03 \cdot b) \oplus (01 \cdot c) \oplus (01 \cdot d)$$

After key xoring the firt byte of the ciphertext will be

$$(02 \cdot a) \oplus (03 \cdot b) \oplus (01 \cdot c) \oplus (01 \cdot d) \oplus k_0$$




$$(02 \cdot a) \oplus (03 \cdot b) \oplus (01 \cdot c) \oplus (01 \cdot d) \oplus 1$$

where $k_0 = 1$ is the first byte of the key. [2 marks]

- **Case 2**: In the second case, we XOR the key bytes first so, the entries of the first column after key xoring wil be

  $$a \oplus k_0, b \oplus k_1, c \oplus k_2, d \oplus k_3$$

  So, now after the mix column the first byte of the ciphertext will be

  $$(02 \cdot (a \oplus k_0)) \oplus (03 \cdot (b \oplus k_1)) \oplus (01 \cdot (c \oplus k_2)) \oplus (01 \cdot (d \oplus k_3))$$

  Now, if all the keys are 1, then it is easy to see that the value of the first byte in this case is also

  $$(02 \cdot a) \oplus (03 \cdot b) \oplus (01 \cdot c) \oplus (01 \cdot d) \oplus 02 \oplus 03 \oplus 01 \oplus 01$$

  $$(02 \cdot a) \oplus (03 \cdot b) \oplus (01 \cdot c) \oplus (01 \cdot d) \oplus 1$$

  [2 marks]

  Similarly, we can show the same for all the bytes

You will only get marks if you have shown this complete computation of the Mix column operation. Without this complete computation you will not be given more than 2 marks in total.