Indian Institute of Technology (IIT-Bombay)

SPRING Semester, 2026
COMPUTER SCIENCE AND ENGINEERING

CS6102: Implementation Security In Cryptography

Quiz- I

Full Marks: 15

Time allowed: 0.75 hours

INSTRUCTIONS: Do not make any extra assumptions other than those stated in the questions. **The marks will only be given if a precise, sound, and clearly written explanation is provided that is understandable by the examiners.** Any extra unnecessary writing, verbal arguments based on that, and unnecessary crib will result in a reduction of 5 marks.

Roll Number: _____

Name: _____

1. **(Easy)** Consider the polynomial $A(x) = \sum_{i=0}^{9} a_i x^i$, $a_i \in \text{GF}(2)$, and a reduction polynomial $P(x) = x^{10} + x^3 + 1$. Compute $C(x) = (A(x))^2 \text{mod} P(x)$. Write down the expressions for the coefficients of $C(x)$ in terms of the coefficients of $A(x)$ [i.e., expression for each c_i, $i \in \{0, 1, \cdots 9\}$]. All calculations have to be shown. No marks without showing proper steps. (5 marks)

Solution:

$$
\begin{aligned}
c_0 &= a_0 + a_5, \\
c_1 &= a_9, \\
c_2 &= a_1 + a_6, \\
c_3 &= a_5, \\
c_4 &= a_2 + a_7 + a_9, \\
c_5 &= a_6, \\
c_6 &= a_3 + a_8, \\
c_7 &= a_7, \\
c_8 &= a_4 + a_9, \\
c_9 &= a_8.
\end{aligned}
$$

0.5 marks for each coefficient. All the calculations have to be shown. No marks without proper steps being shown.

2. **(Medium)** Recall One-Time-Pad (OTP) which results in perfectly secret encryption. For every message $m \in \mathcal{M} := \{0,1\}^l$, a key $k \in \{0,1\}^l$ is sampled uniformly random, and $c = m \oplus k$ is returned as a

ciphertext. The ciphertext space is $\mathcal{C} = \{0,1\}^l$. A company Optiotp uses this algorithm, but with a minor optimization: *instead of choosing* $k \in \{0,1\}^l$*, they choose* $\tilde{k} \in \{0,1\}^{\frac{l}{2}}$ *uniformly random, and then define* $k = \tilde{k}||(\tilde{k} \oplus \alpha)$*, where* $\alpha = \{1\}^{\frac{l}{2}}$*. We also assume that by some magic, both the communicating parties can generate the same* $\tilde{k}$ *at the same time.* Prove/disprove that this algorithm is perfectly secret. In case you think it is not perfectly secret, show a counterexample. Otherwise, show the proof for perfect secrecy (as described in the slides). (5 marks)

Note: If you have to show perfect secrecy, you need to show similar probabilistic arguments that we used to establish the perfect secrecy of OTP. If you have to show a counterexample, show it by choosing a message(s) and ciphertext(s), for which the perfect secrecy definition is violated. Showing for one single violating case would be enough. Also, try to use the definition

$$Pr[C = c|M = m_0] \neq Pr[C = c|M = m_1]$$

if you want to disprove perfect secrecy. It is relatively easier to use.

Solution: The scheme is not perfectly secret. To see this, first observe that $k = \tilde{k}||(\tilde{k} \oplus \alpha) = \tilde{k}||\overline{\tilde{k}}$. Next, consider two messages $m_0 = 0^l$ and $m_1 = 0^{l/2}||1^{l/2}$, and the ciphertext $c = 0^l$. Now, consider:

$$Pr[C = c|M = m_0] = Pr[C = m_0 \oplus k|M = m_0] = Pr[K = c \oplus m_0] = Pr[K = 0^l] = 0$$

and,

$$Pr[C = c|M = m_1] = Pr[C = m_1 \oplus k|M = m_1] = Pr[K = c \oplus m_1] = Pr[K = 0^{l/2}||1^{l/2}] = \frac{1}{2^{l/2}}$$

Therefore, $\exists m \in \mathcal{M}$ and $c \in \mathcal{C}$ such that:

$$Pr[C = c|M = m_0] \neq Pr[C = c|M = m_1]$$

. Hence, it is not perfectly secret.

1 mark if they identify that the scheme is insecure. Full marks only for the correct probabilistic argument.

3. **(Easy)** Let us consider the AES MixColumns operation:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_1 & b_5 & b_9 & b_{13} \\ b_2 & b_6 & b_{10} & b_{14} \\ b_3 & b_7 & b_{11} & b_{15} \end{bmatrix} = \begin{bmatrix} c_0 & c_4 & c_8 & c_{12} \\ c_1 & c_5 & c_9 & c_{13} \\ c_2 & c_6 & c_{10} & c_{14} \\ c_3 & c_7 & c_{11} & c_{15} \end{bmatrix} \tag{1}$$

Here the matrix

$$B = \begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_1 & b_5 & b_9 & b_{13} \\ b_2 & b_6 & b_{10} & b_{14} \\ b_3 & b_7 & b_{11} & b_{15} \end{bmatrix}$$

is the AES state before MixColumns operation and the matrix

$$C = \begin{bmatrix} c_0 & c_4 & c_8 & c_{12} \\ c_1 & c_5 & c_9 & c_{13} \\ c_2 & c_6 & c_{10} & c_{14} \\ c_3 & c_7 & c_{11} & c_{15} \end{bmatrix}$$

is the state after MixColumns operation. The inverse operation is given as:

$$\begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} c_0 & c_4 & c_8 & c_{12} \\ c_1 & c_5 & c_9 & c_{13} \\ c_2 & c_6 & c_{10} & c_{14} \\ c_3 & c_7 & c_{11} & c_{15} \end{bmatrix} = \begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_1 & b_5 & b_9 & b_{13} \\ b_2 & b_6 & b_{10} & b_{14} \\ b_3 & b_7 & b_{11} & b_{15} \end{bmatrix} \tag{2}$$

Recall that each b_i, c_i are elements of $GF(2^8)$. Also, each constant matrix here consists of elements from $GF(2^8)$.

Suppose you are working in a company where you have been asked to implement the MixColumns operation. **But you are only allowed to use the following multiplications in** $GF(2^8)$:

1. $01 \times x$, where $x \in GF(2^8)$ and $\{01\} \in GF(2^8)$.
2. $02 \times x$, where $x \in GF(2^8)$ and $\{02\} \in GF(2^8)$.
3. $04 \times x$, where $x \in GF(2^8)$ and $\{04\} \in GF(2^8)$.
4. $08 \times x$, where $x \in GF(2^8)$ and $\{08\} \in GF(2^8)$.

Show how to compute the matrix multiplications (MixColumns and Inverse MixColumns) only using these operations. Marks will only be given if you justify every step of your computation. (5 marks)

Hint: Focus on the constant matrices. B and C are not important here.

Solution:

$$\begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} 02 & 02 & 00 & 00 \\ 00 & 02 & 02 & 00 \\ 00 & 00 & 02 & 02 \\ 02 & 00 & 00 & 02 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix} + \begin{pmatrix} 00 & 01 & 01 & 01 \\ 01 & 00 & 01 & 01 \\ 01 & 01 & 00 & 01 \\ 01 & 01 & 01 & 00 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix}.$$

$$\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} + \begin{pmatrix} 08 & 08 & 08 & 08 \\ 08 & 08 & 08 & 08 \\ 08 & 08 & 08 & 08 \\ 08 & 08 & 08 & 08 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix}$$

$$+ \begin{pmatrix} 04 & 00 & 04 & 00 \\ 00 & 04 & 00 & 04 \\ 04 & 00 & 04 & 00 \\ 00 & 04 & 00 & 04 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix}.$$

Now, how do you get these magical factorizations of the matrices? For the first one, it is just the `xtime` operation discussed in the class. For the second one, if you take an XOR between the MixColumns and the Inv-MixColumns matrix, you will only see two elements in that matrix:

$$\begin{pmatrix} 0C & 08 & 0C & 08 \\ 08 & 0C & 08 & 0C \\ 0C & 08 & 0C & 08 \\ 08 & 0C & 08 & 0C \end{pmatrix}$$

You can write $0C = 04 \oplus 08$. The rest is straightforward.

2 marks for MixColumns factorization. 3 marks for Inverse MixColumns factorization. But directly writing the answer will fetch only 1 mark in each case. Explanation is mandatory.