Proofs of Security

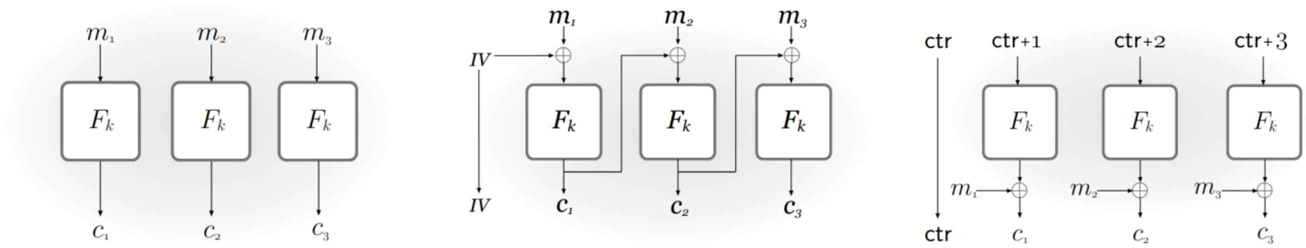
TUT n°6 — Block ciphers

M2-SIF, M2-CRYPTO

Léo Ackermann

Answer all ▸ questions before looking at ⋆ questions.

Block-ciphers modes of operations provide a way of encrypting arbitrary-length messages. Unless stated differently, we consider block ciphers of length n and assume messages to be of length a multiple of n. We recall three of the most common modes*.



ECB mode CBC mode CTR mode

Additionaly, the CBC* mode aims to tackle the inherent sequentiality of CBC mode. For a message $(m_1, \cdots, m_t)$ made of $t \geqslant 2$ blocs the corresponding ciphertext is $(IV, c_1, \cdots, c_t)$ where $\{c_i = E_K(s_i)\}_i, s_0 = IV$ and $\{s_{i+1} = s_i \oplus m_{i+1}\}_i$.

Another important result we recall for this tutorial is the *birthday paradox*.

> **Theorem 1** *For S a finite set of cardinality m, the probability $p(n)$ that a collision occurs when sampling n elements at random from S is $p(n) = 1 - \frac{m!}{(m-n)!} \cdot \frac{1}{m^n}$. It verifies $p(n) \geqslant 1 - e^{-n(n-1)/2m}$. In particular, for $(n-1) \geqslant \sqrt{2\ln(2)m}$ we have $p(n) \geqslant 1/2$.*

Attacking modes

▸ **Question 1.** *Show that ECB mode is not IND-CPA.*

▸ **Question 2.** *Show that CBC* mode is not IND-CPA.*

▸ **Question 3.** *Show that CTR mode does not provide indistinguishability security for long messages. Can we still state that CTR mode is IND-CPA secure?*

▸ **Question 4.** *Show that CBC mode does not provide indistinguishability security for long messages neither.*

⋆ **Question 5.** *Assuming blockciphers working over 64bits, what should be the size of the messages so that the attacks of the questions 3 and 4 succeed with probability greater than 1/2?*

Multiple modes

Multiple modes of operation consists in concatenating modes of operations. For example, the ECB|CBC notation refers to the mode where the output of the ECB mode is the input of the CBC mode. In this exercise, we consider block cipher of length n and of key length l. We assume $n > l$ and that init values IV are known by the adversary.

We first mount a chosen plaintext attack against ECB|ECB|CBC^{-1}. The plaintext P we choose is the concatenation of three n-bits blocks such that $P = (A, A, B)$. The three blocks of the corresponding ciphertexts are denoted (C_1, C_2, C_3).

▸ **Question 6.** *Represent the multiple mode, with its intermediate values A', A'', B' and B''.*

▸ **Question 7.** *Find a relation between A'', k_3, IV and C_1. Find another relation between A'', IV, C_1 and C_2. Deduce a relation between k_3, IV, C_1 and C_2.*

---

*Figures from *Introdctuion to Modern Cryptography*, Katz & Lindell

▸ **Question 8.** *Deduce an attack which recover k_3. How to recover k_1 and k_2 from there? What is the complexity of the whole attack?*

We then mount a chosen ciphertext attack against the CBC|CBC^{-1}|CBC^{-1} mode. We further assume that IV_2 can be programmed (other initial values are fixed and known from the adversary). Consider the following algorithm:

Procedure SearchCollision

1: $i \leftarrow 1$
2: Repeat:
3: Choose $C_1^{(i)}$ and $IV_2^{(i)}$ at random
4: $C_2^{(i)} \leftarrow IV_2^{(i)}$
5: Obtain and store $P_1^{(i)}$ and $P_2^{(i)}$; $i \leftarrow i+1$.
6: until $P_1^{(i)} = P_1^{(j)}$ for some $j < i$, and display the collision.

★ **Question 9.** *Give an approximation of the running time of the former algorithm.*

★ **Question 10.** *Show that if $P_1^{(i)} = P_1^{(j)}$, then $P_2^{(i)} = P_2^{(j)}$.*

★ **Question 11.** *Find a relation between $IV_2^{(i)}$, $IV_2^{(j)}$, k_3, IV_3, $C_1^{(i)}$ and $C_1^{(j)}$ equivalent to $P_1^{(i)} = P_1^{(j)}$. Deduce an attack that recover k_3.*

Padding Oracle attack over CBC

A plaintext is not likely to be exactly of length a multiple of the block size. To bypass this problem, one could use padding: the PKCS7 standard states that the value to pad is the number of bytes that are required. For example, "Hello World" will become "Hello World\5\5\5\5\5" to fit 8-bytes ciphers. We are given an oracle $\mathcal{O}^{\mathsf{Padding}}$ that, given a ciphertext c outputs $\top$ iff the padding of the corresponding plaintext is correct (ie. if it ends with i "$\backslash i$" symbols).

Let $C = (C_1, \cdots, C_N)$ be an intercepted ciphertext. We first focus on the decryption of C_N, the last block of C. Let $C' = (r_1, \cdots, r_n)||C_N$ be a two-blocks long (possibly meaningless) ciphertext forged by the adversary for r_i's of their choice. Let (P_1', P_2') be the corresponding plaintext.

▸ **Question 12.** *Give a relation between P_2', C_{N-1}, $(r_1, \cdots, r_n)$ and P_N the last block of the plaintext corresponding to C.*

▸ **Question 13.** *Assuming that $P_2'[n-1] \neq \backslash 2$, explain how one can recover $P_N[n]$ using $\mathcal{O}^{\mathsf{Padding}}$. How can extra queries to $\mathcal{O}^{\mathsf{Padding}}$ circomvent the need for this assumption?*

▸ **Question 14.** *Show how to set $P_2'[n]$ to $\backslash 2$, and use it to recover $P_N[n-1]$. After explaining why, you may forget about the subtely highlighted in the previous question. Deduce how to recover P_N entirely.*

▸ **Question 15.** *Can we recover all the plaintext blocks?*