TD4 :: Pads and Hashes

Université Sorbonne Paris Nord, M1. Sergey Dovgal
dovgal@lipn.fr

1 Symmetric block cyphers

Vernam's cypher (one-time pads). For a given key k and a plaintext message m of the same length as k, *one-time pad* cypher is defined as $c_k(m) = m \oplus k$.

Exercise 1. Show that without knowing the secret key, it is impossible to recover plaintext from a cyphertext.

Exercise 2. Show that using the same one-time pad several times is a bad idea.

DES encryption (Data Encryption Standard). An encryption of one block of a fixed size 64 bits with a secret key of a fixed size 56 bits is done in several steps.

- The initial plaintext T_0 is split into $L_0||R_0$.
- A *Feistel transform* is defined as follows: if $T_i = L_i||R_i$ is obtained on ith iteration, then $T_{i+1} = L_{i+1}||R_{i+1}$ is given by $L_{i+1} = R_i$ and $R_{i+1} = L_i \oplus f(R_i, k_i)$, where k_i is a secret key, and $f(\cdot, \cdot)$ is a fixed non-linear function.
- The function $f(L, K)$ is described by four steps
  - A linear transform $\widetilde{L} := AL \oplus K$ with a given fixed matrix A.
  - The array $\widetilde{L}$ of length 48 is divided into $\widetilde{L} = \widetilde{L}_0|| \ldots ||\widetilde{L}_7$ blocks of size 6 each.
  - To each array $\widetilde{L}_i$ a boolean function $f_i(\cdot)\colon \{0,1\}^6 \to \{0,1\}^4$ is applied, where each f_i is described by a fixed truth table.
  - The output of f is $f_0(\widetilde{L}_0)|| \ldots ||f_7(\widetilde{L}_7)$.
- The array of keys $k_1, \ldots, k_{16}$, $k_i \in \{0,1\}^{48}$ is obtained by a linear transform of $k \in \{0,1\}^{56}$.

Exercise 3. Describe a decrypting procedure for DES, assuming that the secret key is known.

Exercise 4*. Suggest an encryption procedure for a plaintext m divided into blocks of equal length $m_1||m_2||\dots$ using an encryption function $E(m,k)$ and a secret key k.

Exercise 5* (Double DES). In order to augment the key size of DES, the following encryption block procedure can be suggested: if k_1 and k_2 are two different keys and m is a plaintext message, then the cyphertext is given by $DES_{k_1}(DES_{k_2}(m))$.

- Suggest an algorithm that breaks double DES faster than brute-force search, in time $2^{k_1} + 2^{k_2}$.
- (3-DES) Can you suggest a DES-based scheme using only two keys k_1 and k_2 which is not breakable under the above attack?

2 Hashing functions

The goal is to provide an irreversible hashing function $f\colon \{0,1\}^{32k} \to \{0,1\}^{128}$. It also need to satisfy the property that it is extremely difficult to find two large messages m_1, m_2 such that $f(m_1) = f(m_2)$. Such a situation is called a *collision*.

MD5-hash (broken around 2005 and still widely used in 2020). The states A, B, C, D of 32 bits each are initialised with fixed given values A_0, B_0, C_0, D_0. The states are updated as follows:

$$A_{i+1} = D_i,\ C_{i+1} = B_i,\ D_{i+1} = C_i,$$

$$B_{i+1} = B_i \oplus ((F(B_i, C_i, D_i) \oplus M_i \oplus K_i) \lll s).$$

The function F is fixed for each round $i = 1, \dots, 4$, K_i are fixed constants, and M_i is the plaintext.

Exercise 6. Why MD5 cannot be easily inverted?

Exercise 7* (Birthday paradox). Find a collision in MD5 in 2^{64} operations, instead of brute-force search.

Exercise 8*. Generate two binary files `good` and `evil`, both outputting your name and student card number, adding `This is a GOOD program` or `This it an EVIL program`, and having the same MD5 hash.

Hint: a toolbox for creating programs with an identical `md5sum` hash can be found open-source in the internet.