École normale supérieure **2018-2019**
Département d'informatique

Introduction à la cryptologie
TD n° 5 : Cryptanalyse symétrique

Le système FEAL (pour *Fast Data Encipherment Algorithm*) est un algorithme de chiffrement par bloc qui utilise des clés de 64 bits pour chiffrer des blocs de 64 bits. Il a été proposé par A. Shimizu et S. Miyaguchi en 1987 comme une alternative au système DES. Nous allons étudier une attaque par cryptanalyse différentielle contre sa version originale (appelée $\mathsf{FEAL}-4$) qui est un schéma de Feistel à 4 tours avec pré-blanchiment et post-blanchiment. Le système $\mathsf{FEAL}-4$ est décrit dans la figure 1.

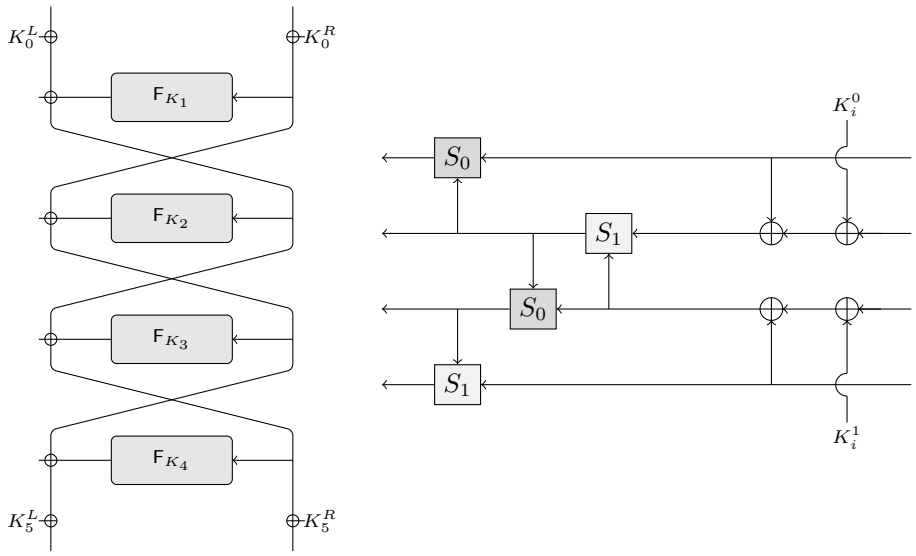


Figure 1 – Description du système de chiffrement par bloc $\mathsf{FEAL}-4$

Après une procédure de diversification de clé, la clé K produit deux sous-clés de 64 bits K_0 et K_5 et quatre sous-clés de 16 bits K_1, K_2, K_3 et K_4. La clé K_0 est utilisée pour le pré-blanchiment (*i.e.* elle est ajoutée au texte clair avant d'appliquer le schéma de Feistel) et la clé K_5 est utilisée pour le post-blanchiment (*i.e.* elle est ajoutée à la sortie du schéma de Feistel pour produire le chiffré).
La fonction de tour F utilise deux S-boîtes S_0 et S_1 définies par

$$S_i(x,y) = (x + y + i \bmod 256) \lll 2 \text{ pour } i \in \{0,1\}$$

où $\lll$ dénote une rotation circulaire sur 8 bits.
En utilisant une clé de tour de 16 bits K_i décomposée en deux octets $K_i = (K_i^{(0)}, K_i^{(1)})$ pour $i \in \{1,2,3,4\}$, la fonction F prenant en entrée 32 bits décomposés en quatre octets $X^i = (x_0^i, x_1^i, x_2^i, x_3^i)$ calcule

$$u_i = S_1(x_0^i \oplus x_1^i \oplus K_i^{(0)}, x_2^i \oplus x_3^i \oplus K_i^{(1)}) \text{ et } v_i = S_0(x_2^i \oplus x_3^i \oplus K_i^{(1)}, u_i)$$

puis retourne

$$F_{K_i}(X) = (S_0(x_0^i, u_i), u_i, v_i, S_1(x_3^i, v_i)).$$

Exercice 1 (Cryptanalyse différentielle de $\mathsf{FEAL}-4$)**.**

1. Montrer que pour tout couple d'octets $(x,y) \in \{0,\ldots,255\}^2$, nous avons

   $$S_0(x \oplus \mathtt{80}, y) = S_0(x,y) \oplus \mathtt{02}$$

   où les constantes 80, 02 sont en notation hexadécimale.
2. Soient $X_0 \in \{0,1\}^{64}$ et $Y_0 = X_0 \oplus (\mathtt{80\ 80\ 00\ 00}, \mathtt{00\ 00\ 00\ 00})$. En notant X_2 et Y_2 leurs images après application de deux tours du chiffrement FEAL avec une clé arbitraire, montrer que

   $$X_2 \oplus Y_2 = (\mathtt{80\ 80\ 00\ 00}, \mathtt{02\ 00\ 00\ 00}).$$

3. En déduire une relation liant les chiffrés de deux messages dont la différence est égale à

$$(\texttt{80 80 00 00}, \texttt{00 00 00 00})$$

qui ne dépend que de la valeur $K_5^R \oplus (0, K_4^0, K_4^1, 0)$.

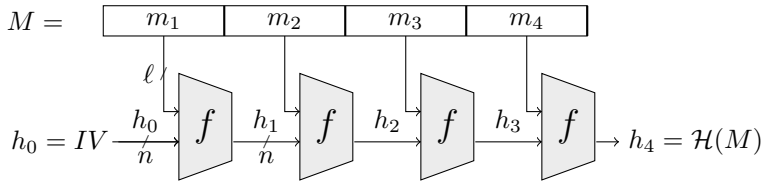
4. Proposer une attaque contre le chiffrement $\mathsf{FEAL}-4$ utilisant seulement deux clairs choisis permettant de retrouver la valeur $K_5^R \oplus (0, K_4^0, K_4^1, 0)$ en 2^{33} évaluations de la fonction F.
5. En supposant connu $K_5^R \oplus (0, K_4^0, K_4^1, 0)$ et en utilisant une propriété différentielle sur un tour, proposer une attaque à deux clairs choisis permettant de retrouver la valeur de $K_5^L \oplus (0, K_3^0, K_3^1, 0)$ en 2^{33} évaluations de la fonction F.
6. Proposer une attaque à huit clairs choisis permettant de trouver une clé équivalente de $\mathsf{FEAL}-4$ en 2^{35} évaluations de la fonction F.

Exercice 2 (Chiffrement DES avec blanchiment)**.** Nous considérons une variante de DES (dite avec *blanchiment*) qui utilise une clé de 120 bits de la forme $K = (K_1, K_2) \in \{0,1\}^{56} \times \{0,1\}^{64}$ et qui chiffre un bloc m de 64 bits sous la forme

$$c = \mathsf{DES}_{K_1}(m) \oplus K_2.$$

Montrer qu'il existe une attaque à deux clairs connus contre cette variante de DES qui demande 2^{57} évaluations de la fonction DES (*i.e.* que cette variante ne ralentit la recherche exhaustive que d'un facteur 2).

Exercice 3 (Multi-collisions pour les fonctions de hachage itérées)**.** Supposons que la fonction de hachage $\mathcal{H} : \{0,1\}^* \to \{0,1\}^n$ est construite à partir d'une fonction de compression $f : \{0,1\}^n \times \{0,1\}^\ell \to \{0,1\}^n$ par la méthode de Merkle-Damgård (avec $\ell \geq n/2$). La construction de Merkle-Damgård est donnée ci-dessous pour un message de longueur 4ℓ. La valeur IV (*initial vector*) est fixe.



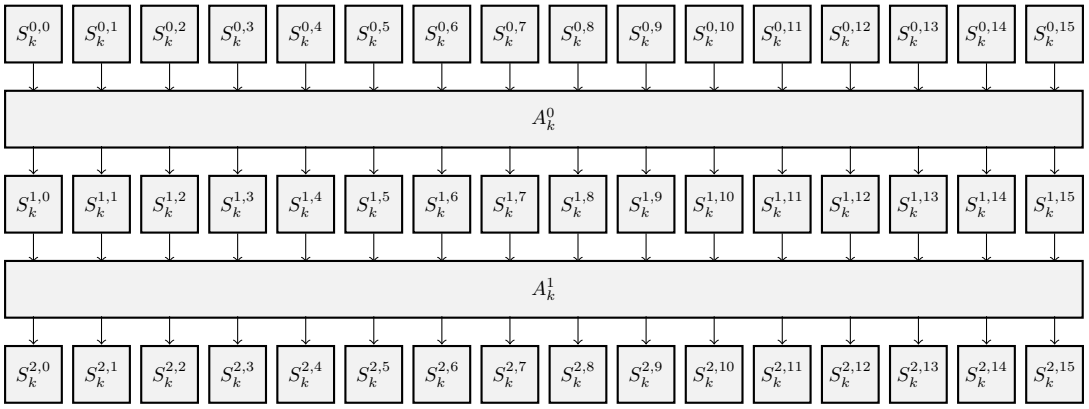
On suppose pour simplifier qu'il n'y a pas de bourrage (*padding*), et que la fonction est utilisée uniquement pour des messages de longueur égale à un multiple de ℓ.

1. Soit M un message quelconque. Évaluer (grossièrement) le nombre d'opérations nécessaires pour trouver de manière générique deux blocs de message distincts $m \neq m'$ tels que $\mathcal{H}(M \parallel m) = \mathcal{H}(M \parallel m')$. Comme « opération » de base on utilisera un appel à la fonction de compression.
2. Soit C le coût en nombre d'opérations de trouver une collision comme dans la question précédente. Montrer comment obtenir une 4-multi-collision pour $\mathcal{H}$, c'est-à-dire trouver 4 messages donnant le même haché, en temps $2C$.
3. Montrer comment obtenir une 2^t-multi-collision pour $\mathcal{H}$ en temps tC.

Supposons que nous avons deux fonctions de hachage $\mathcal{H}$ et $\mathcal{H}'$ avec des sorties de 128 bits. Il est donc possible de trouver des collisions en 2^{64} opérations. Pour éviter cela, une idée naturelle est de concaténer les sorties des deux fonctions, pour créer une fonction de hachage $\mathcal{H}_{\text{super}} : M \mapsto \mathcal{H}(M) \parallel \mathcal{H}'(M)$, dont la sortie est de 256 bits (le symbole $\parallel$ dénote la concaténation de chaînes de caratères).

5. Supposons que $\mathcal{H}$ est construite suivant le schéma de Merkle-Damgård comme précédemment. Conclure quant à la sécurite de $\mathcal{H}_{\text{super}}$: peut-on trouver des collisions en (substantiellement) moins que 2^{128} opérations ?

Exercice 4 (Attaque algébrique sur SASAS)**.** On considère un chiffrement par bloc construit suivant le schéma *Substituton-Permutation Network* (SPN), comme AES, avec 3 couches de boîtes S et deux couches affines intermédiaires. Pour fixer les idées, prenons un taille de bloc de 128 bits, découpés en 16 boîtes S de 8 bits chacune, comme ci-dessous.



On note cette fonction de chiffrement :

$$F_{\text{SASAS}} = S^2 \circ A^1 \circ S^1 \circ A^0 \circ S^0$$

On suppose que tous les composants de ce schéma sont paramétrés par une clef, donc inconnus. De fait on va modéliser toutes les boîtes S et couches affines comme uniformément aléatoires. Supposons qu'en tant qu'attaquant on peut demander des messages clairs choisis. Alors il est possible de retrouver *tous* les composants en temps pratique. Dans cet exercice, notre but va être de retrouver la dernière couche de boîtes S.

Soit I_n l'idéal de $\mathbb{F}_2[X_1, \ldots, X_n]$ généré par $\{X_i^2 - X_i : 1 \leq i \leq n\}$. Soit $\mathbb{F}_2(n)$ le quotient $\mathbb{F}_2[X_1, \ldots, X_n]/I_n$. On identifie les éléments de $\mathbb{F}_2(n)$ avec l'unique représentant dans la classe d'équivalence qui ne contient pas de carré, c'est-à-dire dont tous les monômes sont de la forme $\prod_{i \in S} X_i$ pour un certain $S \subseteq [1, n]$.

1. Soit $P \in F_2(n)$. Montrer que la fonction booléenne $f_P : \{0,1\}^n \to \{0,1\}$ qui à $(x_1, \ldots, x_n)$ associe $P[X_1 = x_1, \ldots, X_2 = x_2]$ est bien définie, c'est-à-dire ne dépend pas du choix du représentant.
2. Montrer que toute fonction $\{0,1\}^n \to \{0,1\}$ est polynomiale. En déduire qu'il y a un isomorphisme entre les fonctions booléennes $\{0,1\}^n \to \{0,1\}$ et les polynômes de $\mathbb{F}_2(n)$.

Dans la suite on identifie fonctions et polynômes. Le degré d'une fonction $\{0,1\}^n \to \{0,1\}^n$ est le max des degrés de chaque bit de sortie (en tant que polynôme sans carré).

3. Soit E un sous-espace affine de dimension $k > 0$ de $\{0,1\}^n$, et P un polynôme de degré $k-1$. Montrer que $\sum_{x \in E} P(x) = 0$.
   Indication. Par linéarité il suffit de montrer cette propriété sur les monômes. Une manière de procéder est : la préimage de 1 par un monôme de degré k est une intersection I de k hyperplans affines ; la somme demandée est égale à la parité de $|E \cap I|$.
4. Montrer que le degré d'une permutation sur n bits est au plus $n-1$.
   Indication. On pourra utiliser la question précédente, et le fait que pour une permutation P, on a $\sum_{x \in \{0,1\}^n} P(x) = 0$.

Soit $\mathcal{M}_a = \{a \parallel x : x \in \{0,1\}^8\}$ pour $a \in \{0,1\}^{120}$ l'ensemble des messages dont les 120 premiers bits sont égaux à a et dont les 8 derniers bits prennent toutes les 2^8 valeurs possibles.

5. Soit $F_{\text{ASAS}} = A^1 \circ S^1 \circ A^0 \circ S^0$ le chiffrement sans la dernière couche S. Montrer $\sum_{x \in \mathcal{M}_a} F_{\text{ASAS}}(x) = 0$.
6. Fixons notre intérêt sur la boîte $S_k^{2,0}$ de la dernière couche S. Soit T son inverse, et soit $X_i = T(i)$ pour $i \in \{0,1\}^8$. Déduire de la question précédente une manière d'obtenir une relation linéaire entre les variables X_i.
7. Conclure en donnant un algorithme permettant de retrouver toutes les boîtes S de la dernière couche.
8. Votre algorithme retrouve-t-il nécessairement les boîtes S du chiffrement original ?