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

Introduction à la cryptologie
TD n° 5 : Cryptanalyse symétrique

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

1. D'une manière générale il suffit de regarder ce qui se passe dans le développement en binaire. Pour justifier en détail on peut procéder en quelques lemmes.

   Lemme 1. $\forall x, (x \oplus \mathtt{80}) \bmod 256 = (x + \mathtt{80}) \bmod 256 = (x \bmod 256) \oplus \mathtt{80}$.

   Démonstration. Dans les trois cas le bit le plus à gauche de x est inversé (modulo 256), et les autres restent inchangés. □

   Lemme 2. $((x \oplus \mathtt{80}) + y) \bmod 256 = ((x + y) \oplus \mathtt{80}) \bmod 256$.

   Démonstration. Immédiat par le lemme précédent, puisque $x \mapsto x \oplus \mathtt{80}$ et $x \mapsto x + \mathtt{80}$ sont identiques modulo 256, et que l'addition commute. □

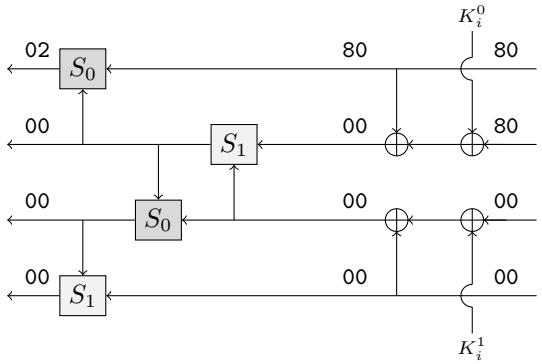
   On déduit :

$$\begin{aligned} S_0(x \oplus \mathtt{80}, y) &= ((x \oplus \mathtt{80}) + y \bmod 256) \lll 2 \\ &= ((x + y) \oplus \mathtt{80} \bmod 256) \lll 2 && \text{Lemme 1} \\ &= ((x + y \bmod 256) \oplus \mathtt{80}) \lll 2 && \text{Lemme 2} \\ &= ((x + y \bmod 256) \lll 2) \oplus (\mathtt{80} \lll 2) \\ &= S_0(x, y) \oplus \mathtt{02} \end{aligned}$$

2. Soient X_1 et Y_1 les images de X_0, Y_0 après un tour de chiffrement. La séquence des différences est :

$$\begin{aligned} X_0 \oplus Y_0 &= (\mathtt{80\ 80\ 00\ 00}, \mathtt{00\ 00\ 00\ 00}). \\ X_1 \oplus Y_1 &= (\mathtt{00\ 00\ 00\ 00}, \mathtt{80\ 80\ 00\ 00}). \\ X_2 \oplus Y_2 &= (\mathtt{80\ 80\ 00\ 00}, \mathtt{02\ 00\ 00\ 00}). \end{aligned}$$

   Pour le voir, il faut regarder le schéma du chiffrement, et observer la propagation des différences. Au premier tour, l'entrée de la fonction de tour F_{K_1} a une différence nulle entre les deux messages, donc sa sortie a également une différence nulle. Lors du second tour, l'entrée de la fonction de tour a une différence $\mathtt{80\ 80\ 00\ 00}$, ce qui donne une différence $\mathtt{02\ 00\ 00\ 00}$ en sortie, comme on le voit ci-dessous.



   Pour la sortie de la boîte S_0 en haut à gauche de ce schéma, on utilise la question précédente.

3. Soient X_0 et Y_0 deux messages clairs dont la différence est

$$X_0 \oplus Y_0 = (\mathtt{80\ 80\ 00\ 00}, \mathtt{00\ 00\ 00\ 00}).$$

Soit $X_i = (X_i^L, X_i^R)$, $Y_i = (Y_i^L, Y_i^R)$ les valeurs respectives de ces messages après i tours de chiffrement ; X_4 et Y_4 sont donc les chiffrés des messages. Notre but est d'exprimer une relation entre X_4 et Y_4. D'après la question précédente, on a :

$$X_2^R \oplus Y_2^R = \texttt{02 00 00 00}.$$

On va exprimer cette relation à partir de X_4, Y_4. Pour cela, il suffit de calculer X_2^R et Y_2^R à partir de X_4, Y_4—ce qui revient à déchiffrer un tour :

$$X_2^R = X_4^L \oplus K_5^L \oplus \mathsf{F}_{K_4}(X_4^R \oplus K_5^R)$$

On remarque par ailleurs que $\mathsf{F}_{(k_1,k_2)}(x) = \mathsf{F}_0(x \oplus (0, k_1, k_2, 0))$, d'où :

$$X_2^R = X_4^L \oplus K_5^L \oplus \mathsf{F}_{K_4}(X_4^R \oplus K_5^R \oplus (0, K_4^0, K_4^1, 0)).$$

De même :

$$Y_2^R = Y_4^L \oplus K_5^L \oplus \mathsf{F}_{K_4}(Y_4^R \oplus K_5^R \oplus (0, K_4^0, K_4^1, 0)).$$

Finalement on obtient la relation :

$$\begin{aligned} & X_4^L \oplus \mathsf{F}_{K_4}(X_4^R \oplus K_5^R \oplus (0, K_4^0, K_4^1, 0)) \\ \oplus\ & Y_4^L \oplus \mathsf{F}_{K_4}(Y_4^R \oplus K_5^R \oplus (0, K_4^0, K_4^1, 0)) \\ =\ & \texttt{02 00 00 00}. \end{aligned}$$

4. On utilise deux messages clairs comme dans la question précédente. On essaie chacune des 2^{32} valeurs possibles de $K_5^R \oplus (0, K_4^0, K_4^1, 0)$ jusqu'à ce que la relation soit vérifiée. Pour la bonne valeur de $K_5^R \oplus (0, K_4^0, K_4^1, 0)$, la relation est nécessairement vérifiée. Comme la relation porte sur 32 bits et qu'on essaie 2^{32} fois, il est possible qu'il y ait des faux positifs, en faible quantité (la probabilité d'avoir un faux positif est heuristiquement la probabilité d'avoir une valeur spécifique parmi $N = 2^{32}$ valeurs, après N tirages, qui tend vers $1 - 1/e$). Chaque essai coûte deux calculs de F, donc le coût total est 2^{33}.
5. Une fois $K_5^R \oplus (0, K_4^0, K_4^1, 0)$ connu, on peut éliminer le dernier tour du chiffrement. En effet, on peut appliquer à chaque chiffré l'inverse du dernier tour de chiffrement, puisque la partie de clef nécessaire est connue. On s'est donc ramené à une construction avec un tour de moins. Au lieu de commencer avec des messages clairs dont la différence est $(\texttt{80 80 00 00}, \texttt{00 00 00 00})$, on peut utiliser deux messages clairs dont la différence est $(\texttt{00 00 00 00}, \texttt{80 80 00 00})$. De cette manière la propagation différentielle des questions précédentes est « remontée » d'un tour, et on peut appliquer la même attaque que précédemment au tour 3, au lieu du tour 4.
6. On peut utiliser la même idée qu'à la question précédente : une fois qu'on a récupéré la clef du 3e tour, on peut l'utiliser pour éliminer ce tour et se ramener à un Feistel à deux tours. On peut procéder comme ci-dessus et commencer avec une différence $(\texttt{80 80 00 00}, \texttt{02 00 00 00})$, ce qui remonte le chemin différentielle encore d'un tour par rapport à la question précédente. On récupère la clef du deuxième tour. Pour le dernier tour il ne reste plus que 2^{32} bits de clef dans la fonction de tour : on peut procéder par force brute.

   Comme on a affaire à un schéma de Feistel à deux tours, des attaques génériques sont aussi possibles.

Exercice 2 (Chiffrement DES avec blanchiment)**.** Soient m, m' deux messages connus, et c, c' les chiffrés correspondants. On remarque $\mathsf{DES}_{K_1}(m) \oplus \mathsf{DES}_{K_1}(m') = c \oplus c'$. On teste donc toutes les valeurs de K_1 jusqu'à ce que la relation soit vérifiée. Comme la relation porte sur 64 bits et qu'on teste une clef de 56 bits, la probabilité de faux positif est négligeable. Chaque essai coûte deux chiffrements DES donc le coût total est 2^{57} chiffrements DES.

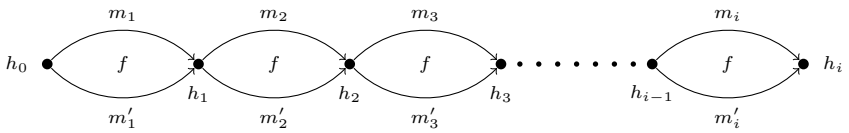
Exercice 3 (Multi-collisions pour les fonctions de hachage itérées)**.**

1. Si on tire N blocs de message m uniformément aléatoirement, par le paradoxe des anniversaires, la probabilité de collision est non-négligeable lorsque N devient proche de $2^{n/2}$. Trouver une telle collision va donc nous coûter de l'ordre de $2^{n/2}$ appels à f. Par ailleurs comme $\ell \geq n$ la probabilité de ne pas trouver une telle paire est négligeable.

2. On génère une première collision arbitraire : on trouve deux blocs de message m_1 et m'_1 distincts dont le haché est en collision. Puis on trouve deux blocs $m_2 \neq m'_2$ tels que $\mathcal{H}(m_1 \parallel m_2) = \mathcal{H}(m_1 \parallel m'_2)$. Pour le prix de trouver deux collisions, obtient ainsi quatre messages en collision : m_1m_2, $m_1m'_2$, m'_1m_2 et $m'_1m'_2$. Le point crucial est que $\mathcal{H}(a \parallel b)$ ne dépend que de $\mathcal{H}(a)$ et de b.
3. On itère le principe de la question précédente. À l'étape i, étant donnés $m_1, \ldots, m_i$ et $m'_1, \ldots, m'_i$ on trouve $m_{i+1} \neq m'_{i+1}$ tels que :

$$\mathcal{H}(m_1 \parallel \cdots \parallel m_i \parallel m_{i+1}) = \mathcal{H}(m_1 \parallel \cdots \parallel m_i \parallel m'_{i+1}).$$

Si on regarde la valeur des hachés intermédiaires, on peut représenter la situation comme suit :



On obtient ainsi 2^i messages en collision en temps iC.

4. Par la question précédente on trouve 2^{64} messages en collision pour $\mathcal{H}$ en temps $64 \cdot 2^{64}$. On calcule l'image par $\mathcal{H}'$ de tous ces messages. Avec bonne probabilité deux d'entre eux sont en collision pour $\mathcal{H}'$. Ils sont donc automatiquement en collision pour $\mathcal{H}_{\text{super}}$. Le coût est de l'ordre de $64 \cdot 2^{64} = 2^{70} \ll 2^{128}$ hachages. Le gain en nombre de bits de sécurité est seulement un facteur logarithmique.

Exercice 4 (Attaque algébrique sur SASAS)**.**

1. Dans $\mathbb{F}_2$ on a clairement $x^2 = x$ donc si on considère le morphisme qui envoie un polynôme sur la fonction correspondante, l'image de l'idéal I_n est identiquement nulle. L'idéal est donc inclus dans le noyau.
2. Réciproquement, comme on est dans un espace fini toute fonction est polynomiale : il suffit de prendre l'interpolation de Lagrange sur tous les points. Plus explicitement, pour montrer la propriété « toute fonction est polynomiale », par linéarité il suffit de le montrer pour une base des fonctions booléennes, donc il suffit de le montrer pour les fonctions du type :

$$\begin{aligned} f_c : \{0,1\}^n &\to \{0,1\} \\ x &\mapsto \begin{cases} 1 \text{ si } x = c \\ 0 \text{ sinon} \end{cases} \end{aligned}$$

pour $c = (c_1, \ldots, c_n) \in \{0,1\}^n$. Or cette fonction est bien polynomiale :

$$f_c(x_1, \ldots, x_n) = \prod (x_i \oplus c_i \oplus 1).$$

3. Par linéarité, on peut se limiter au cas où P est un monôme $X_{i_1} \ldots X_{i_k}$. Soit $S = P^{-1}(\{1\})$ l'ensemble des préimages de 1. Montrer $\sum_{x \in E} P(x) = 0$ revient à montrer que $|S \cap E|$ est pair. Pour cela, remarquons que S est l'intersection de k hyperplans affines définis par l'équation $x_{i_j} = 1$, où x_i est la i-ième coordonnée d'un vecteur dans la base canonique de $\{0,1\}^n$. Cette intersection est non nulle, elle est donc de dimension au moins $n - k$. Donc comme $\dim(E) = k + 1$, $S \cap E$ est soit vide, soit de dimension au moins 1. Il s'ensuit que $|S \cap E|$ est soit 0, soit une puissance 2^d pour $d \geq 1$.
4. Un polynôme P (sans carré) est de degré n ssi il comprend le monôme $\prod_{1 \leq i \leq n} X_i$, ssi il s'écrit sous la forme $\prod_{1 \leq i \leq n} X_i + Q$ pour un certain Q de degré $< n$. Dans ce cas, par la question précédente on remarque $\sum_{x \in \{0,1\}^n} Q(x) = 0$ donc $\sum_{x \in \{0,1\}^n} P(x) = 1$, ce qui est impossible si P est le bit de sortie d'une permutation p, puisqu'on a par ailleurs $\sum_{x \in \{0,1\}^n} p(x) = \sum_{x \in \{0,1\}^n} x = 0$.
5. L'image de $\mathcal{M}_a$ par la première couche S est un certain $\mathcal{M}_b$, en particulier c'est un espace affine de dimension 8. Or les couches ASA suivantes sont de degré au plus 7 par la question précédente, donc on obtient $F_{\text{ASAS}}(x) = 0$ avec la question 3.
6. Soit $s_1, \ldots, s_{256}$ les valeurs de la sortie de la boîte S pour le chiffré de chaque message clair de $\mathcal{M}_a$. Alors la question précédente nous donne l'équation $\sum X_{s_i} = 0$.
7. En recommençant la manœuvre de la question précédente pour différents choix de a, on obtient à volonté des équations linéaire sur les X_i. Lorsqu'on a collecté suffisamment d'équations, on peut résoudre le système et retrouver les X_i. On recommence de même pour chaque boîte S de la dernière couche.

8. Non : les boîtes S de la dernière couche ne sont pas définies de manière unique. En effet on peut composer par exemple $S_k^{2,0}$ avec une application affine A_S en entrée, et composer la dernière couche affine A_k^1 du chiffrement avec l'inverse de A_S. Le chiffrement est inchangé, mais on a une boîte S différente (et une couche affine différente). En réalité, on retrouve les boîtes S à une application affine en entrée près. Cela se traduit dans le fait que le système de la question précédente n'est jamais en réalité de degré plein. Cela n'a pas d'importance : la clef du chiffrement n'est pas définie de manière unique par le fonction de chiffrement, mais ce qui compte pour nous est de retrouver une clef équivalente, qui nous permet de chiffrer et déchiffrer à volonté.

   En termes de complexité, nous avons dû amasser environ 2^8 clairs choisis pour créer le système linéaire à résoudre. Résoudre le système se fait en temps négligeable (environ $(2^8)^3 = 2^{24}$ si on fait un pivot de Gauss basique, ce qui est instantané).