Cryptanalyse — M1MA9W06
Responsable : G. Castagnos

Examen — mardi 15 décembre 2015

Durée 3h
Documents non autorisés
Nombre de pages : 4
Les 4 exercices sont indépendants

1 Attaque sur une composition de chiffrement

Soit $\mathsf{E}_\mathsf{K}(m) = c$ un algorithme de chiffrement symétrique par bloc prenant en entrée un clair m de n bits et une clef secrète K de ℓ bits et produisant un chiffré c de n bits. On note D_K l'algorithme de déchiffrement correspondant. À partir de E, on construit un autre algorithme de chiffrement par bloc, Enc, comme suit : la clef secrète est $(\mathrm{K}_1, \mathrm{K}_2)$ avec $\mathrm{K}_1 \neq \mathrm{K}_2$ et $\mathrm{K}_1, \mathrm{K}_2$ de ℓ bits, et le chiffrement d'un message clair m se fait par

$$Enc_{\mathrm{K}_1,\mathrm{K}_2}(m) := \mathsf{E}_{\mathrm{K}_1}(\mathsf{D}_{\mathrm{K}_2}(\mathsf{E}_{\mathrm{K}_1}(m))).$$

Dans la suite, on suppose la clef secrète $\mathrm{K}_1, \mathrm{K}_2$ fixée, et on considère divers scénarios d'attaques sur le chiffrement Enc pour la récupérer.

(a) Dans cette question, on suppose que l'attaquant fait une attaque à clair connu : il connaît un couple (m, c) avec $c = Enc_{\mathrm{K}_1,\mathrm{K}_2}(m)$ qui de plus vérifie $\mathsf{E}_{\mathrm{K}_1}(m) = 0 \dots 0$, où $0 \dots 0$ représente le bloc constitué de n bits 0. Comment l'attaquant peut il retrouver K_1 et K_2 ? Quelle est la complexité en temps de cette attaque ?

(b) Dans cette question, l'attaquant fait une attaque à clairs choisis. En s'inspirant de l'attaque de la question précédente, montrer comment il peut retrouver K_1 et K_2. Quelle est la complexité en temps et en mémoire de cette attaque ? De combien de clairs choisis a-t-il besoin ?

(c) Dans cette question et la suivante, on suppose que l'attaquant fait une attaque à clairs connus : il connaît 2^p couples (m_i, c_i) pour $1 \leqslant i \leqslant 2^p$, avec $c_i = Enc_{\mathrm{K}_1,\mathrm{K}_2}(m_i)$ et les m_i tous distincts. Soit a un bloc de n bits, quelle est la probabilité qu'il existe i, avec $1 \leqslant i \leqslant 2^p$ et $a = \mathsf{E}_{\mathrm{K}_1}(m_i)$?

(d) En déduire, en s'inspirant des questions précédentes, comment l'attaquant peut retrouver K_1 et K_2 connaissant ces couples (m_i, c_i). Quelle est la complexité en temps et en mémoire de cette attaque (en fonction de n, p et ℓ) ?

(e) Déduire de ces attaques la sécurité de l'option 2 du Triple DES.

$\boxed{2}$ Attaque sur un LFSR filtré

On considère un chiffrement à flot de type LFSR filtré. Ce chiffrement à flot utilise un unique LFSR de longueur 128 avec un polynôme de rétroaction public primitif. Au temps $t = 0$, une clef secrète de 128 bits est chargée dans le registre du LFSR. On note $S^{(t)} = (S_0^{(t)}, S_1^{(t)}, \dots, S_{127}^{(t)})$ l'état interne au temps t. Pour tout $t \geqslant 0$, le bit de suite chiffrante z_t au temps t est obtenu en appliquant une fonction de filtrage f sur l'état interne au temps t :

$$z_t := f(S_0^{(t)}, S_1^{(t)}, \dots, S_{127}^{(t)}) := S_{127}^{(t)} + \sum_{i=0}^{62} S_i^{(t)} S_{\alpha(i)}^{(t)} + S_{10}^{(t)} S_{23}^{(t)} S_{32}^{(t)} S_{42}^{(t)} + \prod_{i=0}^{62} S_i^{(t)} +$$
$$+ S_1^{(t)} S_2^{(t)} S_9^{(t)} S_{12}^{(t)} S_{18}^{(t)} S_{20}^{(t)} S_{23}^{(t)} S_{25}^{(t)} S_{26}^{(t)} S_{28}^{(t)} S_{33}^{(t)} S_{38}^{(t)} S_{41}^{(t)} S_{42}^{(t)} S_{51}^{(t)} S_{53}^{(t)} S_{59}^{(t)},$$

où α est une bijection de $\{0, \dots, 62\}$ dans $\{63, \dots, 125\}$. Puis le registre du LFSR est mis à jour de manière usuelle : $S_i^{(t+1)} = S_{i+1}^{(t)}$, pour $0 \leqslant i \leqslant 126$, et $S_{127}^{(t+1)}$ est mis à jour par une combinaison linéaire dans $\mathbf{F}_2$ de $S_0^{(t)}, S_1^{(t)}, \dots, S_{127}^{(t)}$, en fonction du polynôme de rétroaction.

(a) Expliquer comment Alice et Bob peuvent utiliser ce générateur afin d'échanger N bits de manière confidentielle.

(b) Dans cette question, on suppose qu'un attaquant connaît le bit de suite chiffrante au temps t, z_t. Montrer que s'il connaît de plus 63 bits de l'état interne au temps t (préciser lesquels), alors il peut écrire une équation linéaire dont les inconnues sont les autres bits du registre au temps t.

(c) Déduire de la question précédente une attaque permettant de retrouver la clef secrète, en supposant connu de l'attaquant seulement 33 bits consécutifs de suite chiffrante. Quelle est la complexité de cette attaque ?

(d) Que vaut $f(S_0^{(t)}, S_1^{(t)}, \dots, S_{127}^{(t)})(1 + S_{23}^{(t)})$? Montrer que l'on peut obtenir une expression similaire en multipliant f par $(1 + S_i^{(t)})$ pour un certain entier i (à préciser) avec $i \neq 23$ et $0 \leqslant i \leqslant 127$.

(e) En déduire une attaque plus performante que la précédente contre ce chiffrement à flot visant à retrouver la clef secrète. Donner la complexité de cette attaque et le nombre de bits de suite chiffrante nécessaires pour la mettre en œuvre.

$\boxed{3}$ Attaque sur un système à clef publique

On considère le chiffrement à clef publique suivant. Pour construire ses clefs, Bob choisit un grand entier $q > 0$, et deux entiers f et g tels que $f < \sqrt{q/2}$, $\sqrt{q/4} < g < \sqrt{q/2}$ et $pgcd(f, q) = pgcd(f, g) = 1$. Il calcule ensuite $h \equiv f^{-1}g \ (mod\ q)$ avec $0 < h < q$. Sa clef publique est (h, q) et sa clef privée est (f, g).

Pour envoyer à Bob un message clair m, un entier avec $0 < m < \sqrt{q/4}$, Alice choisit un entier r aléatoire avec $0 < r < \sqrt{q/2}$ et calcule le chiffré $c \equiv m + rh \ (mod\ q)$.

(a) Montrer comment Bob peut déchiffrer c au moyen de sa clef privée.

(b) Montrer de manière informelle comment utiliser le réseau $\mathscr{L}$ de $\mathbf{R}^2$ de base

$$\begin{pmatrix} 1 & h \\ 0 & q \end{pmatrix}$$

pour définir un algorithme polynomial en la taille de q qui retrouve la clef privée à partir de la clef publique.

$\boxed{4}$ Attaque différentielle

Dans cet exercice, on note comme d'habitude par $||$ la concaténation de deux chaînes de bits, et par $\oplus$ l'addition bit à bit modulo 2 de deux chaînes de bits.

On considère un chiffrement par bloc de 64 bits employant 2 clefs de tours K_0 et K_5 de 64 bits et 4 sous clefs de 16 bits, K_1, K_2, K_3 et K_4. Soit F_{K_i} avec $1 \leqslant i \leqslant 4$ une fonction de tour définie plus bas, prenant en entrée 32 bits et ressortant 32 bits. Soit M un bloc de 64 bits à chiffrer. On pose $X_0 = M \oplus K_0$, puis on effectue 4 tours de schéma de Feistel avec la fonction F_{K_i} : On note $X_0 = L_0||R_0$ avec L_0 et R_0 de 32 bits, puis pour $i \in \{1, \dots 4\}$,

$$L_i = R_{i-1}, \qquad R_i = L_{i-1} \oplus F_{K_i}(R_{i-1})$$

Le chiffré est $C = (R_4||L_4) \oplus K_5$.

La fonction de tour utilise deux boîtes S, S_0, S_1 prenant en entrée 16 bits et ressortant 8 bits définie comme suit :

$$S_i(x, y) = (x + y + i \mod 256) << 2,$$

pour $i \in \{0, 1\}$, en identifiant les entiers entre 0 et 255 et leur représentation binaire sur 8 bits (bit de poids faible à droite), et où $<< 2$ désigne la rotation circulaire de deux bits vers la gauche.

La fonction de tour $F_K(X)$ pour X de 32 bits et K une sous clef de 16 bits est définie ainsi : on note $X = x_0||x_1||x_2||x_3$ avec les x_i de 8 bits, et $K = K^L||K^R$ avec K^L, K^R de 8 bits. On calcule :

$$u = S_1(x_0 \oplus x_1 \oplus K^L, x_2 \oplus x_3 \oplus K^R),$$

et

$$v = S_0(x_2 \oplus x_3 \oplus K^R, u)$$

Enfin $F_K(X) = S_0(x_0, u)||u||v||S_1(x_3, v)$.

(a) Montrer que pour tout $(x, y) \in \{0, 1\}^8 \times \{0, 1\}^8$,

$$S_0(x \oplus 1000\,0000, y) = S_0(x, y) \oplus 0000\,0010.$$

16

(b) Soit $M, M^\star \in \{0, 1\}^{64}$ deux messages clairs, tel que $M \oplus M^\star = 1000\,0000\,1000\,0000 \dots 0000$. On note $(L_2||R_2)$ (resp. $(L_2^\star||R_2^\star)$) l'entrée du troisième tour lors du chiffrement de M (resp. de $M^\star$).

Que vaut la différence à l'entrée du troisième tour, c'est à dire $(L_2||R_2) \oplus (L_2^\star||R_2^\star)$?

(c) Soit $O = 0 \ldots 0$ la chaîne nulle de 16 bits. Soit $K = K^L || K^R$ une sous clef de 16 bits avec K^L, K^R de 8 bits. Montrer que pour tout X de 32 bits,

$$F_K(X) = F_O(X \oplus (0000\,0000 || K^L || K^R || 0000\,0000)).$$

(d) On note $K_5 = K_5^L || K_5^R$ avec K_5^L, K_5^R de 32 bits.

Déduire des deux questions précédentes une attaque utilisant 2 clairs choisis permettant de retrouver la valeur de $K_5^R \oplus (0000\,0000 || K_4^L || K_4^R || 0000\,0000)$. Quelle est sa complexité ?