Cryptographie - Master Ingénierie Informatique - Examen 2016-2017

Consignes. Durée 2h00. Tout document ou dispositif électronique est interdit. Le barème est donné à titre indicatif.

Exercice 1 (générateur affine) (4 points) *Le texte chiffré* $c = 111\ 101\ 100\ 101\ 101\ 010\ 111$ *a été obtenu en effectuant un xor du texte clair* p *avec une suite* z *de bits générés en utilisant un générateur basé sur la congruence linéaire*

$$(z_{i+1} \equiv az_i + b)\ mod\ 8$$

Les entiers z_i *ainsi générés sont interprétés comme des nombres à trois bits* $b_{i2}b_{i1}b_{i0}$ *de façon telle que* $z_i = b_{i2} \cdot 2^2 + b_{i1} \cdot 2 + b_{i0}$. *Ainsi on aura:*

$$\begin{aligned} z &= b_{02}b_{01}b_{00}\ b_{12}b_{11}b_{10}\ \cdots\ b_{62}b_{61}b_{60} \\ c &= p \oplus z \end{aligned}$$

On connaît les premiers 9 bits du texte clair p*:*

$$p = 100\ 111\ 001\ \cdots$$

Déterminez le reste du texte clair.

Exercice 2 (CPA-confidentialité et MAC) (6 points) **Rappel:** *On dit qu'un MAC assure l'intégrité si un attaquant PPT ayant pris connaissance des MAC des messages* $m_1, \ldots, m_p$ *(*p *polynomial en* n*) a une probabilité* **négligeable** *de produire un MAC* **valide** *pour un message* m *différent de* $m_1, \ldots, m_p$. *Soit* $\mathcal{F} : 2^n \to [2^n \to 2^n]$ *un générateur de fonctions pseudo-aléatoire (PRF).*

1. *On suppose devoir chiffrer des messages composés de* $(3 \cdot n)$ *bits. Décrivez une méthode de chiffrement basée sur le PRF* $\mathcal{F}$ *qui assure la CPA-confidentialité.*

2. *On suppose maintenant devoir assurer la CPA-confidentialité* **et** *l'intégrité de messages* $m, m', \ldots$ *composés de* n *bits. Décrivez une méthode basée sur le PRF* $\mathcal{F}$ *qui assure ces propriétés.*

3. *On suppose associer à un message* m *de* n *bits la valeur* $E_k(m)$ *définie par :*

$$E_k(m) = [r \leftarrow 2^n : (r, \mathcal{F}(k)(r) \oplus m)]\ .$$

   Est-ce suffisant pour assurer la CPA-confidentialité **et/ou** *l'intégrité du message ?*

4. *Supposons que l'attaquant observe que le calcul de la fonction (probabiliste)* E_k *sur un message* m *(connu par l'attaquant) produit le couple* (r, y). *Par ailleurs, l'attaquant observe aussi que le calcul de la fonction (probabiliste)* E_k *sur un message* m' *(pas connu par l'attaquant) produit le couple* (r, y'). *Dans ce cas, l'attaquant peut-il calculer* m' *?*

Exercice 3 (chiffrement RSA) (6 points)

1. *Programmez une fonction* Java *d'en tête:*

   public static int racinecub(int x)

   qui prend en entrée un entier positif x *et qui retourne (s'il existe) un entier* y *tel que* $y^3 = x$ *et un entier* -1 *autrement. Il s'agit donc de calculer (si elle existe) la racine cubique dans* $\mathbf{Z}$ *(et non pas dans* $\mathbf{Z}_n$ *!). Votre programme devrait être linéaire dans le nombre de bits nécessaires à représenter* x*.*

2. *Rappelez les principes du chiffrement hybride à clef publique.*

Supposons que pour implémenter le chiffrement hybride on dispose d'un système de chiffrement symétrique avec clef de 128 *bits et d'un système RSA avec module* $n = p \cdot q$ *de* 512 *bits.*

3. *Sous quelles hypothèses peut-on choisir un exposant de chiffrement* $e = 3$ *?*

4. *Supposons que pour le chiffrement RSA on choisit bien un exposant de chiffrement* $e = 3$ *et que pour le chiffrement hybride on regarde une clef symétrique* k *comme un entier dans l'intervalle* $[0, 2^{128} - 1]$*. Montrez que dans ce cas un attaquant a une méthode efficace pour déchiffrer les messages sans connaître la clef de déchiffrement.*

Exercice 4 (Diffie-Hellman en groupe) (4 points) *Les participants sont identifiés par un élément de* $\mathbf{Z}_n = \{0, \ldots, n-1\}$*. Les participants souhaitent établir une clef commune. Pour chaque couple de participants* $(i, j) \in \mathbf{Z}_n^2$ *on dispose d'un canal de communication authentifié mais qui n'assure pas la confidentialité des messages échangés. Les participants appliquent une généralisation du protocole de* **Diffie-Hellman**.

- *Le participant* 0 *communique à chaque participant un (grand) nombre premier* p *et un générateur* g *pour* $(\mathbf{Z}_p)^*$*.*
- *Chaque participant choisit de façon indépendante et avec probabilité uniforme un élément* $a_i \in \{2, \ldots, p-2\}$*.*
- *A la fin du protocole, la clef commune est:*

$$K = A_{\{0,\ldots,n-1\}} = (g^{(\Pi_{i \in \{0,\ldots,n-1\}} a_i)}) \bmod p$$

1. *Proposez un protocole d'échange de valeurs de la forme* (I, A_I)*:*

$$(A_I \equiv g^{(\Pi_{i \in I} a_i)}) \bmod p \qquad I \subset \{0, \ldots, n-1\}$$

   avec les objectifs suivants :

   - *Permettre à chaque participant de calculer la clef* K*.*
   - *Éviter que l'observation des valeurs échangées* (I, A_I) *permette à un attaquant de calculer la clef commune* K*.*

   Vous devez expliciter les échanges de valeurs pour le cas où $n = 2$ *(trois participants).*