Master 2ème année – SeCReTS

Examen "Cryptologie industrielle"

27 novembre 2019

Consignes :

- Durée : 2h.
- Documents interdits. Aucun accès à un téléphone portable, une calculatrice, un PDA ou tout autre dispositif électronique, connectable ou non.

Exercice 1. *Fonctions de hachage*

1. Rappeler la définition d'une fonction à sens unique, d'une fonction à collisions faibles difficiles et d'une fonction à collisions fortes difficiles.

2. Soit une fonction de hachage h donnant en sortie une chaîne de n bits. On considère k messages choisis aléatoirement, notés $x_1, x_2, \ldots, x_k$, et on note $y_1 = h(x_1)$, $y_2 = h(x_2)$, $\ldots$, $y_k = h(x_k)$. Retrouver, en fonction de n et k, la probabilité P que ces k messages donnent une collision sur la sortie de n bits.

3. Soit $h : \{0,1\}^* \longrightarrow \{0,1\}^n$ une fonction de hachage (qui est en particulier à collisions faibles difficiles, et à collisions fortes difficiles). Soit $h' : \{0,1\}^* \longrightarrow \{0,1\}^{n+1}$ définie par :

$$h'(x) = \begin{cases} 0||x & \text{si } x \in \{0,1\}^n \\ 1||h(x) & \text{sinon} \end{cases}$$

   Montrer que h' n'est pas à sens unique, mais est encore à collisions faibles difficiles et à collisions fortes difficiles

4. Une méthode classique pour construire une fonction de hachage $h : \{0,1\}^* \longrightarrow \{0,1\}^n$ consiste à itérer une *fonction de compression* $f : \{0,1\}^\ell \longrightarrow \{0,1\}^n$, avec $\ell > n$. L'idée est de décomposer le message m en blocs $m_1, m_2, \ldots, m_k$ de longueur $\ell - n$, puis de poser $y_0 =$IV pour une valeur IV$\in \{0,1\}^n$, puis

$$y_i = f(y_{i-1}||m_i) \qquad (1 \leq i \leq k)$$

   On pose alors $h(m) = y_k$. Pour que la longueur de m soit divisible par $\ell - n$, on utilise un schéma de "padding" standard.

   Soit $g : \{0,1\}^n \times \{0,1\}^n \longrightarrow \{0,1\}^n$ un algorithme de chiffrement par blocs, avec une clé de longueur n bits, et une taille de bloc de n bits (la clé est le premier paramètre, le bloc est le second paramètre). On définit alors une fonction de compression $f : \{0,1\}^{2n} \longrightarrow \{0,1\}^n$ par :

$$f(y||m) = g(y,m)$$

   Montrer comment on peut trouver une collision pour la fonction h.

Exercice 2. *Courbes elliptiques*

On considère la courbe elliptique $y^2 = x^3 + 4x + 1$ sur $(\mathbb{Z}/5\mathbb{Z}, +, \cdot)$.

1. Montrer que les paramètres de cette courbe sont tels que la courbe définit un groupe (Abélien).

2. Déterminer les points de la courbe.

3. Montrer que $P = (0, 1)$ est générateur du groupe.

4. Calculer $5P$ au moyen d'un algorithme systématique.

5. Montrer que le groupe défini par la courbe est isomorphe à $(\mathbb{Z}/n\mathbb{Z}, +)$ où n est le nombre de points de la courbe.

Exercice 3. *Diffie-Hellman et authentification*

Alice et Bob veulent se mettre d'accord sur une clé commune, et utilisent pour cela le protocole d'échange de clés de Diffie-Hellman :

- Alice choisit x aléatoire dans l'intervalle $[0, p-1]$ et calcule $X = g^x \bmod p$;
- Alice envoie X à Bob ;
- Bob choisit y aléatoire dans l'intervalle $[0, p-1]$ et calcule $Y = g^y \bmod p$;
- Bob envoie Y à Alice.

1. Quelle est la valeur de la clé commune ainsi échangée ?

2. Faites tourner l'algorithme d'échange de clé de Diffie-Hellman avec les valeurs suivantes :
   - $p = 11$ comme nombre premier ;
   - $g = 2$ comme générateur de $\mathbb{Z}/p\mathbb{Z}$;
   - $x = 4$ comme nombre secret choisi par Alice ;
   - $y = 8$ comme nombre secret pour Bob.

   Détaillez les calculs en mettant en avant les messages échangés par Alice et Bob. Quelle est la clé ainsi obtenue ?

3. Décrire l'attaque dite "par le milieu" dans laquelle un adversaire *actif* (*i.e.*, qui peut modifier les données pendant le protocole Diffie-Hellman) peut ensuite intercepter, déchiffrer et modifier toutes les communications qu'Alice (ou Bob) chiffrera avec sa clé. Expliquer en détails.

Pour parer à cette attaque, on emploie une version *authentifiée* du protocole Diffie-Hellman : les deux messages échangés entre Alice et Bob sont signés par ces derniers (on suppose qu'Alice et Bob disposent chacun d'une paire de clés publique/privée).

4. Décrire exactement les opérations effectuées par Alice et Bob (calculs, messages à envoyer, etc).

Dans le reste de l'exercice, Alice et Bob souhaitent toujours utiliser la version authentifiée de Diffie-Hellman mais n'ont pas de clés publiques. Ils disposent seulement d'un mot de passe commun. Soit f une fonction (publique) permettant de *padder* le mot de passe en une clé plus longue (par exemple 128 bits) et $\mathcal{E}$ un algorithme de chiffrement par blocs.

5. Évaluer l'entropie (nombre de bits) d'un mot de passe de 6 caractères choisis parmi $[0, 9] \cup [a, z] \cup [A, Z]$.

6. Même question pour un mot de passe choisi dans un dictionnaire de 800 000 mots, avec 1 000 variantes par mot.

Alice et Bob utilisent le protocole suivant, dérivé de Diffie-Hellman (pw désigne le mot de passe) :

- Alice choisit x aléatoire dans l'intervalle $[0, p-1]$ et calcule $X = g^x \bmod p$;
- Alice envoie $\mathcal{E}_{f(pw)}(X||Alice)$ à Bob ;
- Bob choisit y aléatoire dans l'intervalle $[0, p-1]$ et calcule $Y = g^y \bmod p$;

- Bob envoie $\mathcal{E}_{f(pw)}(Y||Bob)$ à Alice.

On prend comme principe de base qu'un mot de passe est une donnée de petite taille, et que le dictionnaire des mots de passe est *énumérable* (contrairement à un espace de clés).

7. Décrire une attaque contre le protocole ci-dessus, dans laquelle l'adversaire retrouve le mot de passe (sans supposer que l'adversaire casse le problème mathématique Diffie-Hellman)

8. Alice et Bob peuvent-ils détecter cette attaque de l'adversaire ?

9. Comment modifier le protocole pour éviter cette attaque ?