CRYPTO 1 - TD 5
2024-2025

Énoncé

Exercice 1 :

Plusieurs solutions ont été proposées pour construire des signatures basées sur la primitive RSA qui résistent à toutes les formes de contrefaçon. Elles utilisent généralement une fonction d'encodage $\mathcal{F} : \mathcal{M} \longrightarrow \mathbb{Z}_N^*$ qui casse les propriétés algébriques de la fonction RSA (où $\mathcal{M}$ désigne l'espace des messages à signer).

Génération des clés : Le signataire tire aléatoirement deux nombres premiers p et q et calcule $N = pq$. Il calcule la fonction indicatrice d'Euler de N, $\varphi(N) = (p-1)(q-1)$. Il choisit un exposant public e premier à $\varphi(N)$ et calcule d l'inverse de e modulo $\varphi(N)$. La clé publique est le couple (N, e) et la clé secrète est l'entier d.

Signature : étant donné un message $m \in \mathbb{Z}_N^*$, le signataire calcule la signature $\sigma \equiv \mathcal{F}(m)^d \bmod N$.

Vérification : étant donné un message $m \in \mathbb{Z}_N^*$ et une signature supposée $\sigma \in \mathbb{Z}_N^*$, l'algorithme de vérification accepte σ si et seulement si $\sigma^e \equiv \mathcal{F}(m) \bmod N$.

1.a] Montrer que si la fonction $\mathcal{F}$ est une fonction de hachage qui n'est pas résistante à la pré-image alors le protocole $\mathcal{F}$-RSA n'est pas résistant à la contrefaçon existentielle sous une attaque sans message.

1.b] Montrer que si la fonction $\mathcal{F}$ est une fonction de hachage qui n'est pas résistante à la seconde pré-image alors le protocole $\mathcal{F}$-RSA n'est pas résistant à la contrefaçon universelle sous une attaque à un message choisi.

1.c] Montrer que si la fonction $\mathcal{F}$ est une fonction de hachage qui n'est pas résistante aux collisions alors le protocole $\mathcal{F}$-RSA n'est pas résistant à la contrefaçon existentielle sous une attaque à un message choisi.

Exercice 2 :

En 1984, T. ElGamal a proposé le premier exemple de signature dont la sécurité repose sur le problème du logarithme discret

Génération des clés : Le signataire choisit un nombre premier p et g un générateur de $\mathbb{Z}_p^*$. Il tire uniformément aléatoirement $x \in \mathbb{Z}_{p-1}$ et calcule $y = g^x \bmod p$. La clé publique est (p, g, y) et la clé secrète associée est x.

Signature : Pour signer un message $m \in \mathbb{Z}_{p-1}$, le signataire tire uniformément aléatoirement $k \in \mathbb{Z}_{p-1}^*$ et calcule $r = g^k \bmod p$. Il calcule $s = (m - xr)/k \bmod p - 1$ et la signature est le couple (r, s).

Vérification : Un couple (r, s) est une signature valide de $m \in \mathbb{Z}_{p-1}$ si et seulement si $(r, s) \in \mathbb{Z}_p \times \mathbb{Z}_{p-1}$ et

$$g^m = y^r r^s \bmod p.$$

2.a] Montrer que le protocole de signature d'ElGamal naïf n'est pas résistant aux contrefaçons existentielles sous une attaque sans message.

Exercice 3 :

3.a] Étudier la sécurité du protocole de signature d'ElGamal lorsque le signataire utilise toujours le même couple $[(r = g^k \mod p), k]$ précalculé pour accélérer le calcul des signatures

3.b] Supposons que pour accélérer le calcul des signatures d'ElGamal, le signataire calcule deux couples $[(r = g^k \mod p), k]$ et $[(a = g^\alpha \mod p), \alpha]$ et utilise pour la i-ème signature la clé temporaire $[(r_i = g^{k+i\alpha} \mod p), k + i\alpha]$ générée par une simple multiplication dans $\mathbb{Z}_p^*$. Étudier la sécurité du protocole de signature obtenu.

Exercice 4 :

Une *signature de Lamport* (ou *signature jetable*) est une méthode pour construire un protocole de signature numérique dont la sécurité repose sur une fonction à sens-unique $f : X \longrightarrow Y$.

Génération des clés : étant donnée une fonction à sens unique $f : X \longrightarrow Y$ et un espace de message $\mathcal{M} = \{0,1\}^k$, le signataire tire uniformément aléatoirement $2k$ valeurs

$$(x_1^{(0)}, x_1^{(1)}, x_2^{(0)}, x_2^{(1)}, \ldots, x_k^{(0)}, x_k^{(1)}) \in X^{2k}$$

et calcule, pour $i \in \{1, \ldots, k\}$ et $j \in \{0,1\}$, $y_i^{(j)} = f(x_i^{(j)})$. La clé publique est le vecteur

$$(y_1^{(0)}, y_1^{(1)}, y_2^{(0)}, y_2^{(1)}, \ldots, y_k^{(0)}, y_k^{(1)}) \in Y^{2k}$$

et la clé secrète est le vecteur

$$(x_1^{(0)}, x_1^{(1)}, x_2^{(0)}, x_2^{(1)}, \ldots, x_k^{(0)}, x_k^{(1)}) \in X^{2k}.$$

Signature : Pour signer un message $m = (m_1, \ldots, m_k) \in \mathcal{M}$ où $m_i \in \{0,1\}$ pour $i \in \{1, \ldots, k\}$, le signataire révèle $\sigma = (x_1^{(m_1)}, \ldots, x_k^{(m_k)}) \in X^k$.

Vérification : Le k-uplet $\sigma = (\sigma_1, \ldots, \sigma_k) \in X^k$ est une signature valide de $m = (m_1, \ldots, m_k) \in \mathcal{M}$ où $m_i \in \{0,1\}$ pour $i \in \{1, \ldots, k\}$ pour la clé publique

$$(y_1^{(0)}, y_1^{(1)}, y_2^{(0)}, y_2^{(1)}, \ldots, y_k^{(0)}, y_k^{(1)}) \in Y^{2k},$$

si et seulement si $f(\sigma_i) = y_i^{(m_i)}$ pour tout $i \in \{1, \ldots, k\}$.

4.a] Montrer que le protocole de signature de Lamport ne peut pas être utilisé pour signer un message de longueur arbitraire $\ell \leq k$.

4.b] Proposer une variante du protocole de signature de Lamport qui permet de signer un message de longueur arbitraire $\ell \leq k$ avec une clé publique de taille $O(k + \log k)$.