7. Etude d'un schéma de signature

7.1 Signature RSA de base

Considérons le schéma de signature suivant :
- la clé publique pk de vérification est définie par un module RSA $n = pq$, ainsi qu'un exposant e, premier avec $\varphi(n)$;
- la clé de signature sk consiste en l'exposant $d = e^{-1} \bmod \varphi(n)$.
- la signature d'un message $m \in \mathbb{Z}_n^\star$ est $\sigma = m^d \bmod n$;
- la vérification d'un couple $(m, \sigma) \in \mathbb{Z}_n^\star \times \mathbb{Z}_n^\star$ consiste en le test $\sigma^e \stackrel{?}{=} m \bmod n$.

Q-1. Décrire une falsification existentielle (sans message connu) contre ce schéma, en précisant sa complexité algorithmique et son succès.

Rappel : Le succès $\mathsf{Succ}^{\mathsf{ef}}_{\Sigma}(\mathcal{A})$ d'un attaquant $\mathcal{A}$ pour produire une falsification existentielle contre un schéma de signature $\Sigma = (\mathcal{K}, \mathcal{S}, \mathcal{V})$ est

$$\mathsf{Succ}^{\mathsf{ef}}_{\Sigma}(\mathcal{A}) = \Pr\left[\ (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathcal{K}(1^k), (m, \sigma) \leftarrow \mathcal{A}(\mathsf{pk}) : \mathcal{V}(\mathsf{pk}, m, \sigma) = 1\ \right].$$

7.2 Signature « Hash-and-Invert »

Une méthode classique, qui permet à la fois de contrer cette attaque et de signer efficacement de longs messages, consiste à effectuer l'opération ci-dessus, non plus sur le message m, mais sur un haché $H(m)$ de ce dernier. Cette construction admet une preuve de sécurité, mais en faisant l'hypothèse que H ressemble à une fonction parfaitement aléatoire (*i.e.* dans le modèle de l'oracle aléatoire).

Gennaro, Halevi et Rabin ont proposé une construction différente, dont la sécurité peut être prouvée dans le modèle standard, mais sur une variante plus faible du problème RSA :
- pour le paramètre de sécurité k, on utilisera la fonction commune à toutes les clés publiques f_k. Sa définition et son utilisation seront précisées ultérieurement. L'algorithme de génération de clés produit deux grands nombres premiers p et q, sur k bits ($2^{k-1} < p, q < 2^k$), et calcule $n = pq$. La clé publique pk de vérification est définie par ce module RSA $n = pq$, ainsi que par un élément aléatoire $y \in \mathbb{Z}_n^\star$; La clé de signature sk consiste en la factorisation de n ou, de façon équivalente, $\varphi(n)$.
- la signature d'un message m est la racine e-ième de y, pour e dépendant de m, en fonction de f_k : $e = f_k(m)$, alors on calcule $d = e^{-1} = (f_k(m))^{-1} \bmod \varphi(n)$, puis $\sigma = y^d \bmod n$;
- la vérification d'un couple $(m, \sigma) \in \{0,1\}^\star \times \mathbb{Z}_n^\star$ consiste en le test $\sigma^{f_k(m)} \stackrel{?}{=} y \bmod n$.

Q-2. Montrer que f_k ne peut pas être une fonction quelconque (il suffira d'exhiber des exemples de fonctions avec les attaques associées).

7.3 Sécurité prouvée

On restreint f_k à retourner des valeurs entières strictement supérieures à 1. Puis on définit le problème du *RSA flexible* – $\mathsf{Fl\text{-}RSA}$: étant donnés un module RSA n, ainsi qu'un élément $y \in \mathbb{Z}_n^\star$, fournir un entier $e > 1$ et une racine e-ième de y modulo n.

$$\mathsf{Succ}^{\mathsf{fl-rsa}}(\mathcal{A}) = \Pr\left[\ (n, y) \leftarrow \mathcal{K}(1^k), (e, x) \leftarrow \mathcal{A}(n, y) : (x^e = y \bmod n) \wedge (e > 1)\ \right].$$

Q-3. Exhiber une réduction du cassage de ce problème à une falsification existentielle (sans message connu), et ce, indépendamment du choix de la fonction f_k à valeurs dans $\mathbb{N}^\star$.

7.4 Simulation des signatures

Cette sécurité est malheureusement insuffisante. Elle garantit l'infalsifiabilité, mais à condition de ne révéler aucune signature. Dans un deuxième temps, nous allons étudier les attaques à messages connus : on fournit à l'attaquant la clé publique de vérification, ainsi qu'une liste de ℓ couples message-signature (m_i, σ_i).

Q-4. Supposons des exposants impairs fixés $e_1, \ldots, e_\ell < 2^{k-2}$. Montrer que, à partir d'une instance (n, y) du *problème du RSA Flexible*, on peut générer une nouvelle instance (n, z) telle que
- z soit uniformément distribué dans $\mathbb{Z}_n^\star$, si y est initialement uniformément distribué dans $\mathbb{Z}_n^\star$;
- on connaisse les ℓ racines e_i-ièmes de z modulo n ;
- une nouvelle racine e-ième de z, pour e premier avec le produit E des e_i, nous permette de résoudre l'instance (n, y).

Remarque : On supposera que $n = pq$ est un premier « fort », c'est-à-dire que $p = 2p' + 1$ et $q = 2q' + 1$, avec p' et q' tous les deux premiers, sur $k - 1$ bits. Ainsi, $\varphi(n) = 4p'q' : 2$ est le seul diviseur premier plus petit que 2^{k-2}.

7.5 Hypothèse d'indivisibilité

On restreint désormais f_k à retourner des entiers impairs entre 2 et 2^{k-2} (où k est la taille en bits des facteurs premiers p et q, supposés « forts »), puis à satisfaire la propriété calculatoire suivante de **ℓ-indivisibilité** : étant donnés $x_1, \ldots, x_\ell$, il est difficile de trouver x tel que $f_k(x)$ divise le produit des $f_k(x_i)$.

Q-5. Montrer que l'implémentation de cette signature avec une telle fonction *ℓ-indivisible* résiste aux falsifications existentielles, même selon des attaques à ℓ messages connus, sous l'hypothèse que le problème du RSA flexible est difficile.

7.6 Attaques à messages choisis : fonctions caméléons

Il existe ensuite une conversion générique, pour faire passer le niveau de sécurité de la résistance aux attaques à messages connus à la résistance aux attaques à messages choisis adaptatives (qui donnent accès à l'attaquant à un oracle de signature), en utilisant une fonction caméléon : une fonction $h(m, r)$ qui résiste aux collisions, mais dont une trappe permet de trouver une seconde pré-image (m', r') avec la valeur partielle m' choisie.

Soit $h_N : \{0,1\}^\kappa \times \mathbb{Z}_N^\star \to \mathbb{Z}_N^\star$, $h_N(m, r) = y^m r^E \bmod N$, pour un module RSA N, et E un entier premier supérieur à 2^κ, la trappe étant $D = E^{-1} \bmod \varphi(N)$.

Q-6. Montrer que cette fonction h_N est une fonction caméléon :
- trouver une collision (sans la trappe) permet de casser un problème difficile, que l'on précisera ;
- la trappe permet de trouver, pour (m, r) et m' fixés, r' tel que $h_N(m, r) = h_N(m', r')$.

Q-7. Montrer comment on peut en déduire un schéma de signature (probabiliste) sûr contre les attaques à messages choisis adaptatives. Préciser les hypothèses algorithmiques nécessaires pour ce niveau de sécurité.