Crypto1 - TD 4
2023 – 2024

Énoncé

Exercice 1 :

Un professeur envoie ses notes au secrétariat de l'école par courriel. La clef publique du professeur est $e_1 = 3$ et $N_1 = 33$, celle du secrétariat est $e_2 = 3, N_2 = 55$.

1.a] Déterminer la clef privée du professeur et du secrétariat de l'école.

1.b] On suppose que le professeur chiffre les notes avec la clef RSA du secrétariat. Quel message chiffré correspond à la note 4 ?

1.c] Éxpliquer comment le professeur procède pour faire une signature RSA de ses notes.

1.d] Éxpliquer comment le secrétariat procéde pour vérifier la signature RSA du professeur.

1.e] Calculer la signature RSA de la note 8 avec la clef du professeur.

1.f] La signature 20 est-elle une signature valide du professeur pour la note 16 ?

Exercice 2 :

Un cryptosystème a beau se baser sur des principes mathématiques très forts, il suffit d'une mauvaise utilisation de celui-ci pour que la sécurité escomptée soit mise à mal. C'est ce que nous allons voir avec la signature RSA.

Alice a mis à la disposition du public les clefs publiques N et e du cryptosystème RSA. Elle garde secret l'exposant de déchiffrement d.

Pour signer un document $1 < m < N$, Alice calcule la signature $s_m \equiv m^d \bmod N$ et envoie le couple (m, s_m) à ses interlocuteurs. Ces derniers peuvent alors vérifier l'identité de l'expéditeur du message m en vérifiant que $m \equiv s_m^e \bmod N$.

Soit un message chiffré $c \equiv m^e \bmod N$ pour Alice. L'attaquant Albert obtient c et veut pouvoir retrouver le message de départ m. On suppose qu'Alice utilise les mêmes clefs pour signer et chiffrer ses messages.

2.a] Soient $u \equiv (r^e c)^d \bmod N$, et $t \equiv r^{-1} \bmod N$. Montrer que $tu \equiv m \bmod N$?

2.b] Trouver le bon $x \neq 1$ tel que Albert puisse retrouver le message m en faisant signer xc à Alice.

2.c] Qu'en déduisez-vous sur l'utilisation de RSA ?

Un groupe de k amis ont décidé – pour se faciliter la vie – d'utiliser le même module n mais des exposants de chiffrement $e_1, \ldots, e_k$ différents. Nous allons étudier le cas $k = 2$.

On suppose que l'attaquant Albert connaît les messages chiffrés c_1 et c_2 d'un même message clair m pour des exposants e_1 et e_2 qui sont premiers entre eux.

2.d] Montrer qu'il existe (r, s) telles que $re_1 + se_2 = 1$. Dans la suite, on suppose que $r < 0$.

2.e] En déduire que $(c_1^{-1})^{-r}c_2^s \equiv m^1 \mod N$, avec (r,s) comme dans la question précédente.

2.f] Expliquer comment Albert retrouve facilement m à partir de c_1 et c_2.

2.g] Qu'en déduisez-vous sur l'utilisation de RSA avec un module commun et des exposants e_1 et e_2 premiers entre eux ?

Exercice 3 :

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 $\sigma \equiv \mathcal{F}(m)^d \mod N$.

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 $\sigma^e \equiv \mathcal{F}(m) \mod N$.

3.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.

3.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.

3.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 4 :

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 $y = g^x \mod p$. 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 $r = g^k \mod p$. Il calcule $s = (m - xr)/k \mod p-1$ 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

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

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

Exercice 5 :

5.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

5.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 6 :

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

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

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

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

et la clé secrète est le vecteur

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

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

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

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

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

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

6.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 $O(k + \log k)$.