Examen

Cryptographie à clé publique

Les exercices peuvent être traités indépendamment. La qualité de la rédaction et la précision des raisonnements entreront pour une part importante dans l'appréciation des copies. Les documents et calculatrices sont autorisés. Durée : 2h00

I. Chiffrement multiplicatif

On considère l'anneau $\mathbb{Z}_{30} = \{0, 1, 2, \ldots, 29\}$ des entiers modulo 30.
Rappelons qu'un élément $a \in \mathbb{Z}_{30}$ est inversible si, et seulement si, $\text{pgcd}(a, 30) = 1$.

1. Énumérer tous les éléments de $\mathbb{Z}_{30}^\star$ (les éléments de $\mathbb{Z}_{30}$ inversibles).
2. Calculer l'inverse dans $\mathbb{Z}_{30}^\star$ des éléments trouvés à la question précédente.
3. On définit le procédé de chiffrement multiplicatif sur $\mathbb{Z}_{30}$ de la façon suivante :

Les messages clairs et chiffrés sont des éléments de $\mathbb{Z}_{30}$ et l'espace des clés est donné par $\mathcal{K} = \{a \in \mathbb{Z}_{30} |\ \text{pgcd}(a, 30) = 1\}$. La fonction de chiffrement pour $a \in \mathcal{K}$ est donnée par

$$E_a(x) = ax \mod 30.$$

On suppose que les letters sont codées comme d'habitude par

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	’	.	,	
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29

* apostrophe = 26 point = 27 virgule = 28 caractère blanc (espace) = 29

(a) Calculer le nombre de clés possibles.

(b) Décrire la fonction de déchiffrement D_a pour $a \in \mathcal{K}$.

(c) Chiffrer le message suivant avec la clé $a = 13$ (vous donnerez le résultat sous la forme du texte correspondant à la suite de nombres) :

UN ORNITHORYNQUE TRISTE

(d) Identifier et expliquer quelles sont les vulnerabilités d'un tel cryptosystème.

II. Factorisation

1. En admettant que l'entier 14803 est le produit de deux nombres premiers, pouvez-vous facilement le factoriser ? Expliquez pourquoi.
2. Si en outre, on révèle que $\varphi(14803) = 14560$, la factorisation est-elle possible ? Donnez les deux facteurs.

III. Fonctions de hachage

Le but de l'exercice est de montrer les liens d'implication ou de non-implication parmi les propriétés des fonctions de hachage :

Propriétés

- **Résistance à la préimage** : étant donné h, on ne peut pas trouver en un temps raisonnable de x tels que $h = H(x)$
- **Résistance à la seconde préimage** : étant donné x, on ne peut pas trouver en un temps raisonnable de $y \neq x$ tels que $H(y) = H(x)$
- **Résistance aux collisions** : on ne peut pas trouver en un temps raisonnable de couples x et y tels que $H(x) = H(y)$

1. Montrer une relation entre la résistance à la seconde préimage et la résistance aux collisions d'une fonction de hachage.
2. Considérons la fonction $f : \mathbb{Z} \to \mathbb{Z}_n$ définie par $f(x) = x^2 \mod n$ pour n un module RSA.
   (a) Justifier que la fonction f est résistante à la préimage.
   (b) Justifier que la fonction n'est pas résistance à la seconde préimage
3. Considérons une fonction de hachage $g : \{0,1\}^* \to \{0,1\}^n$ résistante à la collision. Remarquons que g prend en entrée une sequence de n'importe quelle taille et calcule des empreintes de longueur n. Considérons ensuite la fonction de hachage h qui calcule des empreintes de longueur $n+1$ construites de la façon suivante :

$$h : \{0,1\}^* \to \{0,1\}^{n+1}$$

$$h(x) = \begin{cases} 1|x & \text{si } x \text{ est de longueur } n \\ 0|g(x) & \text{sinon} \end{cases}$$

   où | désigne l'opération de concaténation.
   (a) Montrer que cette fonction est résistante à la collision
   (b) Montrer que cette fonction n'est pas résistante à la préimage
4. Donner un exemple de fonction résistante aux collisions et à la seconde préimage, mais pas à la préimage. Est-ce une fonction de hachage ?

IV. Signature RSA

Alice a mis à la disposition du public les clés publiques RSA ($e = 17, n = 437$) du cryptosystème RSA. Elle garde secret l'exposant d.

1. Sans utiliser la clé secrète, générer un couple (m, σ) valide pour un message m quelconque. (Falsification Existentielle)
2. Soient $(m_1, \sigma_1) = (100, 156)$ et $(m_2, \sigma_2) = (2, 257)$ deux documents signés par Alice. Montrer qu'il est facile de fabriquer une signature valide pour le message $m = 200$.
3. Montrer une attaque à clairs choisis qui permet de signer n'importe quel message. (Falsification Universelle)

Algorithmes

- **Génération des clés**
  $\mathsf{pk} : \mathsf{n}, \ \mathsf{e} \pmod{\varphi(\mathsf{n})}$
  $\mathsf{sk} : \mathsf{d} \pmod{\varphi(\mathsf{n})}$
- **Signature**
  $\mathcal{S}(\mathsf{d}, \mathsf{m}) = \sigma$
  $\sigma = m^d \pmod{\mathsf{n}}$
- **Vérification**
  $\mathcal{V}(\mathsf{e}, \mathsf{m}, \sigma) = \mathsf{oui/non}$
  $m \stackrel{?}{=} \sigma^e \pmod{\mathsf{n}}$

V. Preuves de connaisance sans divulgation

Le but de l'exercice est de s'identifier en tant que le possesseur d'une clé publique El Gamal. On rappelle l'algorithme de génération de clés ElGamal :

Clés ElGamal

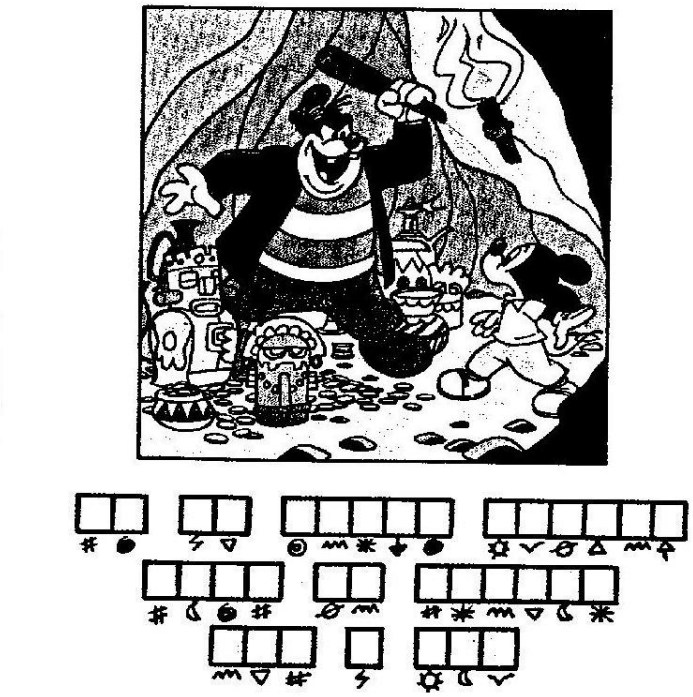
- Soit un premier p et le groupe cyclique $\mathbb{Z}_{\mathsf{p}}^{\star}$
- Soit q un diviseur premier de $(\mathsf{p} - 1)$.
- Soit $\mathsf{g} \in \mathbb{Z}_{\mathsf{p}}^{\star}$ un élément d'ordre q.
- Soit une clé secrète $\mathsf{sk} : \mathsf{x} \pmod{\mathsf{q}}$.
- Soit la clé publique $\mathsf{pk} : \mathsf{y} = \mathsf{g}^{\mathsf{x}} \pmod{\mathsf{p}}$.

Alice prétend être en possession de la clé secrète x associée à la clé publique y.

1. Comment Alice, peut-elle prouver la validité de sa clé publique y sans révéler sa clé secrète x. (Illustrer un protocole de divulgation nulle de connaissance pour le secret x.)
2. Montrer deux moyens differents qui permetent à Alice de tricher dans le protocole précédent.
3. Avec quelle probabilité aura-t-elle la chance de nous faire croire qu'elle posède le secret x après 20 executions du protocole ?

- **Question bonus**

  Que dit Pat à Mickey ?



(Source : Le Journal de Mickey, n 2119, p. 46)