

Cryptographie à clé publique – Feuille de TD 5

04/03/2026

Le corrigé de certains exercices sera disponible à l'adresse suivante :

<https://lvz1.fr/teaching/2025-26/cp.html>

(★) exercice fondamental (★★) pour s'entraîner (★★★) pour aller plus loin  sur machine

Exercice 1. (★) Signature ElGamal : réutilisation de l'aléa.

On considère le schéma de signature ElGamal dans lequel le message est haché avant d'être signé. On choisit comme groupe \mathbb{F}_p^\times , et on a donc à disposition une fonction de hachage $H : \{0, 1\}^* \rightarrow \mathbb{Z}/(p-1)\mathbb{Z}$.

La génération de clés est identique au cadre du cours, mais l'algorithme de signature est légèrement modifié pour donner l'Algorithme 1. Pour la vérification de la signature, on effectue également un haché du message avant de tester l'égalité requise.

Algorithme 1 : Algorithme de signature d'ElGamal avec fonction de hachage

Entrée : un message $m \in \{0, 1\}^*$, la clé privée $a \in \{1, \dots, p-2\}$

Sortie : une signature $s \in \mathbb{F}_p^\times \times \{0, \dots, p-2\}$

- 1 Choisir $k \in \mathbb{Z}/(p-1)\mathbb{Z}$ inversible.
- 2 Calculer $b = g^k \pmod p$.
- 3 Calculer $h = H(m)$.
- 4 Calculer $c = (h - ab)k^{-1} \pmod{(p-1)}$.
- 5 Retourner la signature $s = (b, c)$.

On suppose dans cet exercice qu'Alice réutilise le même aléa k pour plusieurs de ses signatures, et l'on souhaite en déduire une faille de sécurité importante.

Question 1.– Soient $s = (b, c)$ et $s' = (b', c')$ les signatures de deux messages distincts m et m' effectuées avec le même aléa k . Comparer b et b' , puis déterminer une égalité liant $h = H(m)$, $h' = H(m')$, a , b , c et c' .

Question 2.– En déduire une attaque à message connu sur la clé privée d'Alice, qui réussit avec très bonne probabilité.

Exercice 2. (★★) Une proposition de schéma de signature.

Dans cet exercice, on considère un nombre premier p pour lequel le problème du logarithme discret dans \mathbb{F}_p^\times est supposé difficile. On note g un générateur du groupe cyclique \mathbb{F}_p^\times . Enfin, on considère une fonction de hachage $H : \{0,1\}^* \rightarrow \mathbb{Z}/(p-1)\mathbb{Z}$.

Un schéma de signature est décrit par les trois algorithmes suivants.

Algorithme 2 : Génération de clefs

Entrée : les paramètres du système

Sortie : une paire de clefs publique/privée

- 1 Tirer x aléatoirement dans $\mathbb{Z}/(p-1)\mathbb{Z}$.
 - 2 Tirer y aléatoirement dans $\mathbb{Z}/(p-1)\mathbb{Z}$.
 - 3 Calculer $X = g^x \pmod p$ et $Y = g^y \pmod p$.
 - 4 La clef publique est $\text{pk} = (X, Y)$, la clef privée est $\text{sk} = (x, y)$.
-

Algorithme 3 : Signature

Entrée : un message $m \in \{0,1\}^*$, la clé privée $\text{sk} = (x, y)$

Sortie : une signature $s \in \mathbb{Z}/(p-1)\mathbb{Z}$

- 1 Calculer $h = H(m) \in \mathbb{Z}/(p-1)\mathbb{Z}$
 - 2 Calculer et retourner l'élément $s = xh + y \in \mathbb{Z}/(p-1)\mathbb{Z}$.
-

Algorithme 4 : Vérification

Entrée : une signature $s \in \mathbb{Z}/(p-1)\mathbb{Z}$, un message $m \in \{0,1\}^*$, la clé publique $\text{pk} = (X, Y)$

Sortie : vrai ou faux

- 1 Calculer $h = H(m) \in \mathbb{Z}/(p-1)\mathbb{Z}$.
 - 2 Calculer $a = g^s \pmod p$ et $b = X^h Y \pmod p$.
 - 3 Faire le test $a \equiv b \pmod p$ et retourner le booléen associé.
-

Question 1.– Vérifier que le schéma de signature est valide.

Question 2.– Proposer une attaque sur la clé privée $\text{sk} = (x, y)$. On précisera le moyen d'attaque utilisé.

Exercice 3. (★) Signatures et fonctions de hachage.

Soit H une fonction de hachage à valeurs dans $\{0,1\}^t$, où $t \geq 1$ est un entier fixé. On rappelle qu'une *collision* sur H est un couple de messages distincts $m \neq m'$ tels que $H(m) = H(m')$.

Question 1.– Théoriquement, on peut obtenir une collision sur la fonction de hachage H par un compromis temps-mémoire, et exploiter ainsi le *paradoxe des anniversaires*. Décrire la méthode qui permet d'obtenir cette collision, et donner une approximation de sa complexité en fonction de t . Pour cela, on supposera que le coût d'évaluation de H , et le test d'appartenance d'un haché h à une liste de hachés L se font en temps constant.

On considère maintenant le schéma de signature DSA dans le groupe multiplicatif \mathbb{F}_p^\times , et on note g un générateur d'un sous-groupe d'ordre q de \mathbb{F}_p^\times , où q divise $p - 1$. Pour simplifier, on suppose également que la fonction de hachage H est utilisée **sans schéma de remplissage** additionnel. Les algorithmes de signature et de vérification de DSA sont rappelés ci-dessous. On note $\mathcal{S} = (\mathbb{Z}/q\mathbb{Z})^\times \times (\mathbb{Z}/q\mathbb{Z})^\times$ l'espace des signatures.

Algorithme 5 : Signature DSA

Entrée : un message m , la clé privée a

Sortie : une signature $s \in \mathcal{S}$

- 1 Calculer l'entier h associé à $H(m) \in \{0,1\}^t$.
 - 2 Choisir $k \in (\mathbb{Z}/q\mathbb{Z})^\times$ aléatoirement.
 - 3 Calculer $b = (g^k \bmod p) \bmod q$.
 - 4 Calculer $c = (h + ab)k^{-1} \bmod q$.
 - 5 Si b ou c n'est pas inversible $\bmod q$, revenir à l'étape 2.
 - 6 Sinon, retourner $s = (b, c)$.
-

Algorithme 6 : Vérification DSA

Entrée : une signature $s \in \mathcal{S}$, un message m , la clé publique $\alpha = g^a$

Sortie : vrai ou faux

- 1 Calculer l'entier h associé à $H(m) \in \{0,1\}^t$.
 - 2 Calculer $x = g^{hc^{-1} \bmod q} \alpha^{bc^{-1} \bmod q}$.
 - 3 Faire le test $x \equiv b \bmod q$ et retourner le booléen associé.
-

Question 2.– Expliquer comment une collision sur H peut mener à une attaque sur le schéma de signature. On précisera la nature et les moyens de l'attaque.

Question 3.– En déduire la valeur de t minimale pour **espérer** obtenir une sécurité EUF-CMA (infalsifiabilité existentielle à message choisi) de 128 bits.