École normale supérieure **2019-2020**
Département d'informatique

Introduction à la cryptologie
TD n° 7 : Couplages.

Dans toute la suite, on suppose qu'on a un couplage admissible calculable efficacement :

$$e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$$

avec des groupes $\mathbb{G}$, $\mathbb{G}_T$ cycliques d'ordre premier p. Pour g générateur de $\mathbb{G}$, on a donc :
- $g_T = e(g, g) \neq 1$.
- $\forall a, b \in \mathbb{Z}_p, e(g^a, g^b) = g_T^{ab}$.

Exercice 1 (Échauffement)**.**

1. Montrer que les deux propriétés ci-dessus sont vraies pour *un* générateur de $\mathbb{G}$ (au sens : il existe un générateur tel que...) si et seulement si elles sont vraies pour *tout* générateur de $\mathbb{G}$.
2. Montrer que le problème de Diffie-Hellman décisionnel est facile dans $\mathbb{G}$.
(Diffie-Hellman décisionnel : distinguer un triplet (g^a, g^b, g^{ab}) pour a, b uniformes d'un triplet (g^a, g^b, g^c) pour a, b, c uniformes.)

Exercice 2 (Échange de clef sans interaction à trois participants)**.**

1. En s'inspirant du protocole de Diffie-Hellman, décrire un protocole dans lequel trois parties A, B, C se mettent d'accord sur un secret commun en présence d'un adversaire qui observe les communications, et tel qu'il suffit à chaque participant de publier une seule valeur, indépendante des valeurs publiées par les autres participants. On suppose que le problème de Diffie-Hellman bilinéaire (*i.e.* étant donnés g^a, g^b, g^c, calculer g_T^{abc}) est difficile.
2. Supposons maintenant qu'on a une *application multilinéaire* :

$$e_n : \mathbb{G}^n \to \mathbb{G}_T$$
$$(g^{a_1}, \ldots, g^{a_n}) \mapsto e_n(g, \ldots, g)^{a_1 \cdots a_n}$$

où $e_n(g, \ldots, g) \neq 1$. Généraliser l'exercice précédent à $n + 1$ participants. Sur quelle hypothèse repose maintenant la sécurité ?

Exercice 3 (Poignées de main secrètes)**.** France 1789. Une société secrète révolutionnaire souhaite créer un protocole permettant aux membres de la société secrète de se reconnaître entre eux. Si un agent de l'état essaie de suivre le protocole (sans avoir l'information secrète nécessaire), cependant, il ne doit rien apprendre : il ne doit en particulier pas apprendre si la personne avec qui il interagit fait partie ou non de la société secrète.
La Société tire un nombre aléatoire secret $s \in \mathbb{Z}_p$. On suppose que chaque membre a un identifiant $\mathsf{id} \in \mathbb{G}$, et reçoit secrètement de la société la valeur id^s. Deux membres A et B, d'identifiants respectifs id_A et id_B, peuvent se reconnaître entre eux comme suit. Soit H une fonction de hachage.

a. A et B tirent un nombre à usage unique (*nonce*) aléatoire, respectivement n_A et $n_B \in E$. A envoie n_A et id_A. B envoie n_B et id_B.

b. A envoie $H(e(\mathsf{id}_A^s, \mathsf{id}_B) \parallel n_A \parallel n_B \parallel 0)$. B envoie $H(e(\mathsf{id}_A, \mathsf{id}_B^s) \parallel n_A \parallel n_B \parallel 1)$.

1. Comment les participants peuvent-ils vérifier que ce que l'autre a envoyé est correct ?
2. Quel souci(s) y a-t-il si on omet les nonces ?

3. Une fois que deux membres de la société se sont reconnus, comment peuvent-ils établir une clef secrète commune (sans nouvel échange) ?

Exercice 4 (Sécurité des signatures de Boneh-Boyen)**.** On définit le schéma de signature suivant. La clef secrète est $x \in \mathbb{Z}_p^*$, tiré uniformément aléatoirement. La clef publique est $u = g^x$.

- **Signature :** soit un message $m \in \mathbb{Z}_p$. On suppose $m \neq x$. La signature de m est $\sigma = g^{(x+m)^{-1}}$, où l'inversion est modulo p.
- **Vérification :** étant donnée une signature σ, on vérifier $e(\sigma, ug^m) = g_T$.

On définit le problème q**-Strong Diffie-Hellman** (q-SDH) : étant donné $(g, g^x, g^{x^2}, \ldots, g^{x^q})$ pour un x uniforme, calculer une paire $(m, g^{(x+m)^{-1}})$ (pour un m quelconque). On suppose que ce problème est difficile (pour tout q).

1. Vérifier que l'algorithme de vérification accepte une signature légitime.
2. Montrer qu'on peut utiliser le couplage e pour vérifier qu'une instance $(g, g^x, g^{x^2}, \ldots, g^{x^q})$ de q-SDH est correctement formée. Montrer de même que la version décisionnelle du problème (vérifier si une solution de q-SDH est correcte) est facile.
3. Dans la suite, nous considérons la sécurité de la signature ci-dessus contre un attaquant ayant accès à q messages connus $m_1, \ldots, m_q$ fixés d'avance, et essayant de forger une signature pour un message de son choix m^* distincts des m_i. Nous allons réduire cette sécurité à l'hypothèse q-SDH. On suppose donc qu'on a accès à un attaquant qui réussit à casser la signature au sens précédent (pour tout choix de g), et que nous avons une instance $(g, g^x, g^{x^2}, \ldots, g^{x^q})$ de q-SDH. Notre but est d'utiliser cet attaquant pour résoudre l'instance de q-SDH.
   (a) Dans un premier temps, supposons que nous connaissons magiquement x. On donne à notre attaquant une instance de l'algorithme de signature, où le générateur de $\mathbb{G}$ est :
   $$h = g^{\prod(x+m_i)}.$$
   Expliquer comment on fournit à l'attaquant les signatures des messages m_i, et comment on utilise sa sortie pour résoudre q-SDH.
   (b) Maintenant on ne suppose plus que x est magiquement connu. Montrer comment on peut quand même calculer h en utilisant l'instance q-SDH.
   (c) De même, montrer comment on peut extraire la réponse à l'instance q-SDH de la sortie de l'attaquant (sans connaissance magique de x).

Exercice 5 (Chiffrement basé sur l'identité sans couplage)**.** Considérons le protocole de chiffrement basé sur l'identité suivant.

Setup : l'autorité choisit un module RSA $N = pq$, où $p \equiv q \equiv 3 \mod 4$ sont premiers. L'espace des clairs est $\mathcal{M} = \{-1, 1\}$ et l'espace des chiffrés est $\mathcal{C} = \mathbb{Z}_N$. On suppose qu'on sait publiquement hacher uniformément vers les éléments x tels que $\left(\frac{x}{N}\right) = 1$, et aussi vers les éléments tels que $\left(\frac{x}{N}\right) = 1$. En particulier on associe publiquement à chaque identité un certain a tel que $\left(\frac{a}{N}\right) = 1$.

Extract : la clef secrète de l'utilisateur a est $r = a^{(\phi(N)+4)/8} \mod N$.

Encrypt : Pour chiffrer un bit m (codé dans $\mathcal{M} = \{-1, 1\}$) pour une identité a, un utilisateur :

(a) tire uniformément aléatoirement $t_1 \in \mathbb{Z}_N$ avec $m = \left(\frac{t_1}{N}\right)$;
(b) tire uniformément aléatoirement $t_2 \in \mathbb{Z}_N$ avec $m = \left(\frac{t_2}{N}\right)$;
(c) calcule $c_1 = t_1 + at_1^{-1} \mod N$ et $c_2 = t_2 - at_2^{-1} \mod N$, et retourne le chiffré $s = (c_1, c_2)$.

1. Montrer que puisque $\left(\frac{a}{N}\right) = 1$, soit a, soit $-a$ est un résidu quadratique modulo N.
2. Montrer que dans l'algorithme **Extract**, la clef secrète r retournée par l'autorité satisfait $r^2 = a \mod N$ ou $r^2 = -a \mod N$.
3. Donner l'algorithme de déchiffrement (qui utilisera c_1 et r si $r^2 = a \mod N$ et c_2 et r dans le cas contraire). **Indication.** Observer que $c_1 + 2r$ est égal à un carré multiplié par t_1^{-1}.