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

Introduction à la cryptologie
TD n° 4 : Log discret, signatures

Exercice 1. Soit $\mathbb{G} = \langle g \rangle$ un groupe cyclique d'ordre p premier noté multiplicativement. On suppose le générateur g et p publiques. Le paramètre par rapport auquel on dit qu'un algorithme est « polynomial » est $\log p$. La multiplication dans $\mathbb{G}$ compte comme une unité.

1. Montrer qu'on sait calculer une inversion dans $\mathbb{G}$ en temps polynomial. Indication : $x^p = 1$.
2. Montrer qu'on sait calculer une racine carrée dans $\mathbb{G}$ en temps polynomial. Indication : $x^{p+1} = x$.
3. Qu'en est-il pour une racine n-ième pour n quelconque ? Indication : $\mathbb{G} \sim (\mathbb{Z}_p, +)$.
4. Peut-on calculer une racine cubique en temps polynomial dans le groupe RSA $\mathbb{Z}_{pq}$ pour p, q premiers ? Comment cela se concilie-t-il avec la question précédente ?

Considérons :
- le problème CDH : étant données g^a, g^b pour a, b quelconques, calculer g^{ab}.
- le problème SQUARE : étant donnée g^a, pour a quelconque, calculer g^{a^2}.

Savoir résoudre un problème signifie qu'on sait le résoudre en temps polynomial.

5. Montrer que si on sait résoudre CDH, on sait résoudre SQUARE.
6. Montrer que si on sait résoudre SQUARE, on sait résoudre CDH. Indication : $(a+b)^2 = a^2 + b^2 + 2ab$.
7. L'équivalence reste-t-elle vraie pour le problème CUBE : étant donnée g^a, pour a quelconque, calculer g^{a^3} ? Indication : on peut essayer de se ramener à SQUARE.
8. Même question pour le problème d-CUBE : étant donnée g^a, pour a quelconque, calculer g^{a^d}.

Exercice 2 (Auto-réductibilité du problème du logarithme discret)**.** Soit $\mathbb{G}$ un groupe fini cyclique d'ordre p et g un générateur de $\mathbb{G}$. Considérons una algorithme $\mathcal{A}$ qui prend en entrée un élément de $\mathbb{G}$ et retourne un entier, en temps τ (dans le pire des cas) où τ représente au moins le coût d'une exponentiation dans $\mathbb{G}$.
Supposons qu'il existe un sous-ensemble E de $\mathbb{G}$ avec $|E| \geq \epsilon|\mathbb{G}|$ et $\epsilon \in]0, 1]$ pour lequel lorsque $\mathcal{A}$ est exécuté sur un élément $h \in E$, l'entier retourné par $\mathcal{A}$ est le logarithme discret de h en base g. Considérons l'algorithme $\mathcal{B}$ défini à partir de $\mathcal{A}$ dans l'algorithme 1.

Algorithme 1 Algorithme $\mathcal{B}$

Entrée: $g, h \in \mathbb{G}$
Sortie: $x \in \mathbb{Z}_p$ tel que $h = g^x$.
  tant que VRAI **faire**
    $c \leftarrow_\$ \mathbb{Z}_p$ (c est tiré uniformément aléatoirement dans $\mathbb{Z}_p$)
    $h' \leftarrow g^c$
    $w \leftarrow \mathcal{A}(h \cdot h')$
    si $g^w = h \cdot h'$ **alors**
      retourner $w - c \bmod p$
    fin si
  fin tant que

1. Montrer que l'algorithme $\mathcal{B}$ résout le problème du logarithme discret dans $\mathbb{G}$ en temps espéré $O(\tau/\epsilon)$.
2. Réciproquement, montrer que la distribution uniforme est la « plus difficile » pour le log discret, au sens suivant. Supposons qu'il existe un algorithme $\mathcal{A}'$ tel que si h est tiré uniformément dans $\mathbb{G}$, alors $\mathcal{A}(h)$ calcule le log discret de h avec probabilité p. Soit $\mathcal{D}$ une distribution quelconque sur $\mathbb{G}$. Montrer qu'il existe un algorithme $\mathcal{B}'$ tel que si on tire h suivant $\mathcal{D}$, alors $\mathcal{B}(h)$ calcule le log discret de h avec la même probabilité de succès et le même temps de calcul que $\mathcal{A}'$, plus quelques opérations de base dans $\mathbb{G}$.

Exercice 3 (Logarithme discret de petit poids de Hamming)**.** Le nombre de multiplications effectuées par l'algorithme d'exponentiation dichotomique dépend du nombre de 1 dans le développement en base 2 de l'exposant considéré. Il a donc été suggéré pour rendre les protocoles cryptographiques plus efficaces d'utiliser des clés secrètes où ce nombre, le *poids de Hamming* de l'exposant, est relativement petit. Pour se prémunir d'attaque par recherche exhaustive de la clé secrète, il est nécessaire que le nombre de tels exposants soit suffisamment

grand. Le nombre d'exposants de ℓ bits de poids w est donné par le coefficient binomial $\binom{\ell}{w}$. L'exercice montre une adaptation de l'algorithme de Shanks pour résoudre le problème du logarithme discret de petit poids de Hamming en $O(\ell\binom{\lceil \ell/2 \rceil}{\lceil w/2 \rceil})$ exponentiations dans le groupe considéré.
Considérons un groupe multiplicatif cyclique $\mathbb{G}$ engendré par $g \in \mathbb{G}$ d'ordre connu q un nombre premier de ℓ bits (*i.e.* $2^{\ell-1} < q < 2^\ell$). Soit w un entier dans $\{1, \ldots, \ell\}$. Nous supposerons que ℓ et w sont pairs.

1. Donner un algorithme pour calculer le logarithme discret dans $\mathbb{G}$ d'un élément h dont le poids de Hamming du logarithme discret est égal à w en $O(\binom{\ell}{w/2})$ exponentiations dans le groupe et dont la complexité en mémoire est $O(\binom{\ell}{w/2})$ éléments de groupe. Question bonus : peut-on éviter de payer ce coût en mémoire ?

Soient N un entier et ℓ et w deux entiers pairs. Un *système de décomposition* de type (N, ℓ, w) est un couple $(X, \mathcal{B})$ tel que
— X est un ensemble de cardinal ℓ ;
— $\mathcal{C}$ est une famille de N sous-ensembles de X de cardinaux $\ell/2$;
— pour tout sous ensemble A de X de cardinal w, il existe un sous-ensemble $C \in \mathcal{C}$ tel que $|A \cap C| = w/2$.

2. Montrer que s'il existe un système de décomposition de type (N, ℓ, w) alors le problème du logarithme discret de poids de Hamming égal à w peut être résolu en $O(N\binom{\ell/2}{w/2})$ exponentiations dans le groupe et en stockant $O(\binom{\ell/2}{w/2})$ éléments de groupe.
3. Posons $X = \mathbb{Z}_\ell$. Montrer que la famille $\mathcal{C} = \{C_i, 0 \leq i \leq \ell/2 - 1\}$, définie par

$$C_i = \{i + j \bmod \ell, 0 \leq j \leq \ell/2 - 1\} \quad \text{pour } 0 \leq i \leq \ell/2 - 1$$

   est un système de décomposition de type $(\ell/2, \ell, w)$.
4. Conclure.

Exercice 4 (Logarithme discret calculatoire et décisionnel)**.** Soit $p > 2$ un nombre premier tel que $q = 2p + 1$ est premier également. Soit $\mathbb{G} = \mathbb{Z}_q^*$. On considère un élément h d'ordre p dans $\mathbb{G}$. Soit $\mathbb{H} = \langle h \rangle$.

1. On dit qu'un élément f de $\mathbb{G}$ est un carré ssi : $\exists g \in \mathbb{G}, f = g^2$. Donner un algorithme qui étant donné un élément $f \in \mathbb{G}$ renvoie 1 si f est un carré, 0 sinon.
   Indication : on peut considérer la fonction sur $\mathbb{G}$ qui à x associe x^p.
2. Un élément h d'ordre p existe-t-il nécessairement ? Donner un algorithme pour trouver un tel g.
3. Supposons que nous avons un algorithme polynomial pour résoudre le logarithme discret sur $\mathbb{H}$. En déduire un algorithme polynomial pour résoudre le logarithme discret sur $\mathbb{G}$.
4. Soit A la distribution des triplets (g^a, g^b, g^{ab}) où a, b, sont tirés uniformément aléatoirement dans $\mathbb{Z}_q^*$. Soit B la distribution des triplets (g^a, g^b, g^c) où a, b, c sont tirés uniformément aléatoirement dans $\mathbb{Z}_q^*$. Le problème décisionnel de Diffie-Hellman sur $\mathbb{G}$ est de distinguer ces deux distributions. Donner un algorithme polynomial qui prend en entrée un triplet dans $\mathbb{G}$ et sort un bit, tel que que la différence entre la probabilité que ce bit soit 1 sur une entrée tirée dans A et une entrée tirée dans B est minorée par une constante non nulle.

Exercice 5 (Fonction de hachage et sécurité des signatures de Schnorr)**.** Soit $\mathbb{G}$ un groupe cyclique d'ordre q. Soit $\mathcal{H}$ une fonction de hachage. On rappelle le schéma de signature de Schnorr :

Génération des paramètres : Alice tire une clef secrète $x \in \mathbb{Z}_q^*$ uniformément aléatoirement. Elle publie la clef publique $y = g^x$.

Signature : Pour signer un message M, Alice tire $k \in \mathbb{Z}_q^*$ uniformément aléatoirement. La signature de M est $\sigma = (s, h)$ avec $h = \mathcal{H}(g^k, M)$ et $s = k - xh$.

Vérification : Étant donnés M et $\sigma = (s, h)$, on vérifie $\mathcal{H}(g^s y^h, M) = h$.

1. Montrer que ce schéma de signature est correct.
2. Montrer que si l'espace des messages à signer est $\mathbb{G}$ et qu'on choisit comme fonction de hachage :

$$\mathcal{H} : (M, r) \mapsto M \cdot r$$

   alors le protocole de Schnorr n'est pas résistant aux contrefaçons existentielles.