Travaux Dirigés

Algorithme "Baby-Step Giant-Step"

Soit G un groupe, noté multiplicativement, et $g \in G$ un élément d'ordre n connu. Le but de cet exerice est de présenter et étudier un algorithme "générique" de calcul de logarithme discret en base g : l'algorithme "Baby-Step Giant-Step", inventé par D. Shanks en 1971.

On note h l'élément dont on veut calculer le logarithme discret.

- Calculer $d = \lfloor \sqrt{n} \rfloor$ (partie entière).
- Étape "pas de bébé" : pour tout $i \in \{0, \ldots, d-1\}$ calculer $g^i \in G$ et stocker le couple (g^i, i).
- Étape "pas de géant" : pour j partant de 0,
  - calculer $h.(g^{-d})^j$
  - rechercher dans les valeurs stockées s'il existe un couple (g^i, i) tel que $g^i = h.(g^{-d})^j$:
    si oui, renvoyer $i + dj$,
    si non, incrémenter j.

Terminaison.

1. On suppose qu'il existe un entier s tel que $g^s = h$.
   (a) Justifier pourquoi on peut supposer $s \in \{0, 1, \ldots, n-1\}$.
   (b) En déduire qu'il existe $a, b \in \mathbb{N}$ tels que $s = ad + b$ avec $b \leq d-1$ et $a < n/d$.
2. Expliquer pourquoi la deuxième boucle ("pas de géant") est effectuée au plus n/d fois. Que peut-on dire si l'algorithme n'a rien renvoyé après n/d étapes ?

Analyse de complexité.

3. Quelle est la complexité en mémoire de cet algorithme ?
4. Expliquer comment réaliser la première étape en $O(\sqrt{n})$ opérations dans G.

L'algorithme tel qu'il est présenté ne précise pas, pour la dernière étape, comment rechercher un couple (g^i, i) qui conviendrait.

5. Utilisation de listes triées.
   (a) Expliquer à quelle condition cette recherche peut se faire en $O(\ln(\sqrt{n}))$ opérations. Quelle étape faut-il alors rajouter à l'algorithme ?
   (b) Montrer que la complexité temporelle globale est alors en $O(\sqrt{n}\ln(\sqrt{n}))$.
6. Montrer que l'on peut descendre à $O(\sqrt{n})$ en utilisant des tables de hachage.

Challenge.

8. Soit $p = 400\,001\,201$. Utiliser cet algorithme pour calculer dans $(\mathbb{Z}/p\mathbb{Z})^*$ le logarithme discret en base $g = 4444$ de $h = 349$, sachant que l'ordre de g est $n = 1\,000\,003$.

Un protocole Zero-Knowledge

Un protocole à divulgation nulle de connaissance (*zero-knowledge*) permet à un participant de démontrer qu'il connaît un certain secret, *sans dévoiler aucune information sur ce secret* (ce qui peut paraître paradoxal au premier abord !). Un tel protocole peut servir à des fins d'identification ou d'authentification.

Soient g et h deux éléments d'un groupe G dans lequel le DLP est difficile. Le protocole de Schnorr permet à Pascal (le prouveur) de démontrer à Véronique (la vérificatrice) qu'il connaît le logarithme discret a de h en base g, sans le divulguer. On suppose connu l'ordre n, premier, de g.

— Pascal choisit en entier k au hasard entre 2 et $n-1$ de façon équiprobable, calcule un *engagement* (commitment) $c = g^k$, et l'envoie à Véronique.
— Véronique transmet en retour un *challenge* de son choix s_1 entre 0 et $n-1$.
— Pascal répond en envoyant $s_2 = k + as_1 \bmod n$, où a est le logarithme de h en base g.
— Véronique teste si $g^{s_2} = c.h^{s_1}$ ou pas.

Correction du protocole.

1. Montrer qu'on a bien $g^{s_2} = c.h^{s_1}$ si Pascal a suivi les consignes et connaît le logarithme discret a.

Véronique n'apprend rien.
On suppose que Pascal suit les consignes (et connaît a, évidemment).

2. Montrer que Véronique n'a aucune information sur k au moment où elle choisit s_1
3. En déduire que la valeur de s_2 ne lui apporte aucune information sur a.
4. Pourquoi est-il important que k soit bien aléatoire ?

Pascal ne peut pas tricher.
On suppose ici que Pascal *ne connaît pas* a, mais tente quand même de convaincre Véronique. Il n'est bien sûr pas tenu de respecter les consignes ; il doit cependant envoyer un engagement $c \in G$, qu'il choisit comme il veut.
On va montrer que si s_1 a été choisi équiprobablement, alors la probabilité que la réponse s_2 de Pascal passe la vérification de Véronique est inférieure ou égale à $1/n$.

5. Soient $s_1 \neq s_1'$ deux challenges potentiels de Véronique. Démontrer par l'absurde que Pascal n'est pas capable de fournir des réponses corrects s_2 et s_2' à ces deux challenges (sinon il connaîtrait a).
6. En déduire que Pascal ne peut fournir une réponse correcte qu'à au plus un des challenges possibles de Véronique.
7. Conclure.

Applications.

8. Expliquer comment ce protocole peut servir à identifier Pascal, en présence d'une autorité de confiance.