11. Le chiffrement linéaire

11.1 Les courbes elliptiques et les couplages

Sur certaines courbes elliptiques (dont on notera $(\mathbb{G}, +)$ un sous-groupe d'ordre premier q, engendré par un point P), il existe une application $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, où $\mathbb{G}_T$ est un sous-groupe du groupe multiplicatif (d'ordre q) d'une extension finie d'un corps fini, telle que :

— e est efficacement calculable ;

— e est bilinéaire de $\mathbb{G} \times \mathbb{G}$ dans $\mathbb{G}_T$: pour tous $P, Q \in \mathbb{G}$ et $a, b \in \mathbb{Z}_q$, $e(a \cdot P, b \cdot Q) = e(P, Q)^{ab} \in \mathbb{G}_T$;

— e est non-dégénérée : $e(P, P) \neq 1$.

Note On utilisera donc une notation additive pour le groupe $\mathbb{G}$ et une notation multiplicative pour le groupe $\mathbb{G}_T$.

On peut définir les problèmes Diffie-Hellman usuels :

— le problème Diffie-Hellman calculatoire dans le groupe $\mathbb{G}$ —noté $\mathsf{CDH}_{\mathbb{G}}$— qui consiste à calculer $ab \cdot P$, à partir de $P \in \mathbb{G}$, $P_a = a \cdot P$, $P_b = b \cdot P$, pour $a, b \in \mathbb{Z}_q$. On mesure le succès d'un attaquant $\mathcal{A}$ par

$$\mathsf{Succ}^{\mathsf{cdh}}_{\mathbb{G}}(\mathcal{A}) = \Pr_{\substack{P \in \mathbb{G} \\ a,b \in \mathbb{Z}_q}} [\mathcal{A}(P, a \cdot P, b \cdot P) = ab \cdot P].$$

— le problème Diffie-Hellman décisionnel dans le groupe $\mathbb{G}$ —noté $\mathsf{DDH}_{\mathbb{G}}$— qui consiste à décider si un élément $Q \in \mathbb{G}$ donné est le $\mathsf{CDH}_{\mathbb{G}}$ de (P, P_a, P_b), noté $\mathsf{CDH}_{\mathbb{G}}(P, P_a, P_b)$, ou non. On mesure l'avantage d'un distingueur $\mathcal{A}$ par

$$\mathsf{Adv}^{\mathsf{ddh}}_{\mathbb{G}}(\mathcal{A}) = \left| \Pr_{\substack{P \in \mathbb{G} \\ a,b \in \mathbb{Z}_q}} [\mathcal{A}(P, a \cdot P, b \cdot P, ab \cdot P) = 1] - \Pr_{\substack{P \in \mathbb{G} \\ a,b,c \in \mathbb{Z}_q}} [\mathcal{A}(P, a \cdot P, b \cdot P, c \cdot P) = 1] \right|.$$

Pour toutes les mesures de succès ou d'avantages ci-dessus (et à venir), on définit par $\mathsf{Succ}(t)$ ou $\mathsf{Adv}(t)$, les valeurs maximales atteintes pour des attaquants bornés en temps par t.

Q-1. Il existe de tels contextes $(\mathbb{G}, \mathbb{G}_T, e)$ pour lesquels on peut admettre la difficulté du problème Diffie-Hellman calculatoire sur $\mathbb{G}$ (pour tout temps t raisonnable, $\mathsf{Succ}^{\mathsf{cdh}}_{\mathbb{G}}(t)$ est très petit).
Montrer cependant qu'il n'est pas raisonnable de supposer la difficulté du problème Diffie-Hellman décisionnel sur $\mathbb{G}$: on exhibera un distingueur pour le $\mathsf{DDH}_{\mathbb{G}}$.

Outre les problèmes Diffie-Hellman bilinéaires, que nous n'utiliserons pas ici, on définit les problèmes Diffie-Hellman "linéaires" suivants :

— le problème Diffie-Hellman linéaire calculatoire dans $\mathbb{G}$ —noté $\mathsf{CLDH}_{\mathbb{G}}$— qui consiste à calculer, étant donné $(P, Q, R) \in \mathbb{G}^3$ et $(U = a \cdot P, V = b \cdot Q)$ pour $a, b \in \mathbb{Z}_q$, le résultat $W = (a + b) \cdot R$. On mesure le succès d'un attaquant $\mathcal{A}$ par

$$\mathsf{Succ}^{\mathsf{cldh}}_{\mathbb{G}}(\mathcal{A}) = \Pr_{\substack{P,Q,R \in \mathbb{G} \\ a,b \in \mathbb{Z}_q}} [\mathcal{A}(P, Q, R, a \cdot P, b \cdot Q) = (a + b) \cdot R].$$

— le problème Diffie-Hellman linéaire décisionnel dans $\mathbb{G}$ —noté $\mathsf{DLDH}_{\mathbb{G}}$— qui consiste à décider, étant donné $(P, Q, R) \in \mathbb{G}^3$, $(U = a \cdot P, V = b \cdot Q)$ pour $a, b \in \mathbb{Z}_q$ et $W \in \mathbb{G}$, si $W = (a + b) \cdot R$ ou non. On mesure l'avantage d'un distingueur $\mathcal{A}$ par

$$\mathsf{Adv}^{\mathsf{dldh}}_{\mathbb{G}}(\mathcal{A}) = \left| \begin{array}{r} \Pr_{\substack{P,Q,R \in \mathbb{G} \\ a,b \in \mathbb{Z}_q}} [\mathcal{A}(P, Q, R, a \cdot P, b \cdot Q, (a + b) \cdot R) = 1] \\ - \Pr_{\substack{P,Q,R \in \mathbb{G} \\ a,b,c \in \mathbb{Z}_q}} [\mathcal{A}(a \cdot P, b \cdot Q, c \cdot R) = 1] \end{array} \right|.$$

Q-2. Montrer comment résoudre le problème $\mathsf{CDH}_{\mathbb{G}}$ à partir d'un algorithme qui résoud le problème $\mathsf{CLDH}_{\mathbb{G}}$, en temps t avec probabilité ε.

Q-3. Montrer comment décider le problème $\mathsf{DDH}_{\mathbb{G}}$ à partir d'un algorithme qui décide le problème $\mathsf{DLDH}_{\mathbb{G}}$, en temps t avec avantage ε.

Q-4. Discuter la difficulté des problèmes $\mathsf{CLDH}_{\mathbb{G}}$ et $\mathsf{DLDH}_{\mathbb{G}}$ dans des groupes $\mathbb{G}$ équipés d'une application bilinéaire.

11.2 Un schéma de chiffrement

Considérons le schéma de chiffrement $\mathsf{Enc} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ suivant, dans un contexte $(\mathbb{G}, \mathbb{G}_T, e)$ comme défini ci-dessus, avec R un générateur de $\mathbb{G}$, d'ordre premier q.

— $\mathcal{K}(1^k)$: la clé privée est un couple $\mathsf{sk} = (x, y) \in (\mathbb{Z}_q^\star)^2$,
et la clé publique est $\mathsf{pk} = (P = x^{-1} \cdot R, Q = y^{-1} \cdot R)$;

— $\mathcal{E}(\mathsf{pk}, m)$: pour chiffrer un message $m \in \mathbb{G}$, on choisit $a, b \in \mathbb{Z}^\star$, puis on calcule

$$c_1 = a \cdot P, \quad c_2 = b \cdot Q, \quad K = (a+b) \cdot R, \quad c_3 = K + m.$$

Q-5. Expliquer le déchiffrement à partir de $\mathsf{sk} = (x, y)$.

Q-6. Montrer que ce schéma atteint le niveau de sécurité $\mathsf{IND} - \mathsf{CPA}$. On précisera sous quelle hypothèse algorithmique.

Soit un chiffré $(c_1 = a \cdot P, c_2 = b \cdot Q, c_3 = (a+b) \cdot R + m)$. On définit $\pi = a \cdot R$.

Q-7. Montrer que (c_1, c_2, c_3) est un chiffré de 0 (soit donc $c_3 = (a+b) \cdot R$) si et seulement si

$$e(R, c_1) = e(P, \pi) \quad \text{et} \quad e(R, c_2) = e(c_3 - \pi, Q).$$

11.3 Mise en gage

Un protocole de mise en gage permet de s'engager sur une valeur (un bit β) lors de la phase d'engagement, $C = \mathsf{Commit}(\beta)$, sans ne rien révéler sur cette valeur, mais sans pouvoir la changer ultérieurement, lors de l'ouverture, $\mathsf{Open}(C, \beta)$.

Soit R un générateur de $\mathbb{G}$, d'ordre premier q. On considère l'algorithme de génération de paramètres $\mathsf{Gen}_1 \to (P, Q, R, U, V, W)$:

— on choisit aléatoirement $x, y \in \mathbb{Z}_q^\star$, et on définit $P = x^{-1} \cdot R$ et $Q = y^{-1} \cdot R$) ;

— on chiffre R sous $\mathsf{pk} = (P, Q)$:

$$U = a \cdot P, \quad V = b \cdot Q, \quad W = (a+b) \cdot R + R.$$

La mise en gage d'un bit β consiste en le triplet $\mathsf{Commit}(\beta; r, s) = (C_1, C_2, C_3)$

$$C_1 = \beta \cdot U + r \cdot P \quad C_2 = \beta \cdot V + s \cdot Q \quad C_3 = \beta \cdot W + (r+s) \cdot R.$$

L'ouverture consiste en la publication de (β, r, s).

Q-8. Montrer que ce schéma de mise en gage est *perfectly-binding* : une mise en gage ne peut être ouverte que d'une seule manière (même pour un attaquant tout-puissant).

Q-9. Montrer que ce schéma de mise en gage est *computationally-blinding* : la mise en gage ne révèle pas d'information sur le bit engagé, à un attaquant de puissance de calcul bornée (en revanche, un attaquant tout-puissant peut extraire de l'information).

Q-10. Montrer que la connaissance d'une trappe permet d'extraire efficacement β de (C_1, C_2, C_3).

On modifie maintenant l'algorithme d'initialisation en $\mathsf{Gen}_2 \to (P, Q, R, U, V, W)$: Soit R un générateur de $\mathbb{G}$, d'ordre premier q. On considère l'algorithme de génération de paramètres $\mathsf{Gen}_1 \to (P, Q, R, U, V, W)$:

— on choisit aléatoirement $x, y \in \mathbb{Z}_q^\star$, et on définit $P = x^{-1} \cdot R$ et $Q = y^{-1} \cdot R$) ;

— on chiffre 0 (au lieu de R) sous $\mathsf{pk} = (P, Q)$:

$$U = a \cdot P, \quad V = b \cdot Q, \quad W = (a + b) \cdot R + R.$$

Q-11. Expliquer en quoi les deux processus Gen_1 et Gen_2 fournissent des paramètres (P, Q, R, U, V, W) indistingables.

Q-12. Montrer que ce schéma de mise en gage (avec Gen_2) est désormais *perfectly-blinding* : la mise en gage ne révèle aucune information sur le bit engagé (même à un attaquant tout-puissant).

Q-13. Montrer que ce schéma de mise en gage est désormais *computationally-binding* : une mise en gage ne peut être ouverte que d'une seule manière, à moins d'être en mesure de résoudre un problème difficile.

Soient les paramètres (P, Q, R, U, V, W), une mise en gage $(C_1, C_2, C_2) = \mathsf{Commit}(\beta; r, s)$.

Q-14. Montrer que, si (P, Q, R, U, V, W) provient de Gen_1, $\pi = r \cdot R$ est une preuve que $\beta = 0$. On précisera les tests à vérifier.

Q-15. Montrer que, si (P, Q, R, U, V, W) provient de Gen_2, la connaissance de $\mathsf{sk} = (x, y)$ permet de générer une preuve "acceptable" d'engagement de 0, pour toute mise en gage (C_1, C_2, C_3), et donc y compris pour une mise en gage de 1, sans connaître la valeur engagée ni les aléas utilisés.