Master CSI 2 2011-2012

Cryptologie Avancée — M1MA9W07

Responsables : G. Castagnos – G. Zémor

Examen — 20 décembre 2011

Durée 3h — Documents non autorisés

Partie G. Castagnos

1 Une variante d'Elgamal

On considère le schéma de chiffrement asymétrique « Hash Elgamal », défini dans le modèle de l'oracle aléatoire comme suit :

- Soit $\mathcal{H}$ un oracle aléatoire pouvant prendre en entrée des éléments de tout groupe cyclique G retourné par $\mathsf{GenDH}(1^k)$ ci-dessous et retournant des valeurs de $\{0,1\}^k$ pour tout $k \in \mathbf{N}$
- Soit GenDH un algorithme polynomial qui prend en entrée 1^k et retourne la description d'un groupe cyclique G son ordre q premier tel que $|q| = k$ et un générateur g.
- L'algorithme KeyGen appelle GenDH puis choisit x aléatoire avec probabilité uniforme dans $\mathbf{Z}/q\mathbf{Z}$ et calcule $X = g^x$. KeyGen retourne $pk = (G, q, g, X)$ et $sk = (G, q, g, x)$.
- L'algorithme $\mathsf{Encrypt}$ sur l'entrée (pk, m) avec $m \in \{0,1\}^k$ choisit y uniformément dans $\mathbf{Z}/q\mathbf{Z}$ calcule $Y = g^y$ et $Z = X^y$ dans G et retourne $c = (Y, m \oplus \mathcal{H}(Z))$, où $\oplus$ désigne l'addition modulo 2 bit à bit.

(a) Donner l'algorithme de déchiffrement $\mathsf{Decrypt}$ correspondant, montrer que pour tout $m \in \{0,1\}^k$ et tout couple de clefs (pk, sk), $\mathsf{Decrypt}(sk, \mathsf{Encrypt}(pk, m)) = m$.

(b) Quel est l'intérêt de cette variante comparé au chiffrement d'Elgamal classique ?

Dans le reste de l'exercice, on note $\mathcal{A}$ un attaquant polynomial probabiliste $\mathsf{IND-CPA}$ contre Hash Elgamal et on suppose que l'avantage ϵ de $\mathcal{A}$ lors de l'expérience $\mathsf{IND-CPA}$ est non négligeable.

(c) Donner l'expérience $\mathsf{IND-CPA}$ que joue $\mathcal{A}$. Que signifie que son avantage est non négligeable ?

À partir de $\mathcal{A}$, on veut construire un attaquant $\mathcal{B}$ avec un succès non négligeable pour l'expérience ci-après :

On définit l'expérience $\mathbf{Exp}^{\mathsf{List-CDH}}_{\mathsf{GenDH},k}(\mathcal{B})$:

(a) Lancer GenDH avec entrée 1^k pour obtenir G, q, g

(b) Choisir $x, y \xleftarrow{\$} (\mathbf{Z}/q\mathbf{Z})$, et calculer $X = g^x$ et $Y = g^y$

(c) $\mathcal{B}$ prend $G, q, g, (X, Y)$ en entrée et renvoie L un ensemble d'éléments de G

(d) La sortie de l'expérience est 1 si $g^{xy} \in L$ et 0 sinon

Le succès de $\mathcal{B}$ pour résoudre le problème $\mathsf{List-CDH}$ est

$$\Pr\left[\mathbf{Exp}^{\mathsf{List-CDH}}_{\mathsf{GenDH},k}(\mathcal{B}) = 1\right].$$

(d) Donner un algorithme qui permet à $\mathcal{B}$ d'interagir avec $\mathcal{A}$ pour simuler l'expérience $\mathsf{IND-CPA}$.

(e) Comment $\mathcal{B}$ peut-il simuler l'oracle aléatoire $\mathcal{H}$ auquel $\mathcal{A}$ a accès ? En particulier, comment utiliser la liste des requêtes de $\mathcal{A}$ pour résoudre le problème $\mathsf{List-CDH}$?

(f) Donner une minoration de la probabilité de succès, en fonction de ϵ, de l'algorithme $\mathcal{B}$ que vous avez construit pour résoudre $\mathsf{List-CDH}$. Conclure sur la sécurité $\mathsf{IND-CPA}$ d'Hash Elgamal dans le modèle de l'oracle aléatoire.

2 Relation entre $\mathsf{List-CDH}$ et CDH

Dans cet exercice on veut établir les relations qu'il existe entre le problème classique CDH et la version $\mathsf{List-CDH}$ introduite par l'expérience $\mathbf{Exp}^{\mathsf{List-CDH}}_{\mathsf{GenDH},k}(\mathcal{B})$ dans l'exercice précédent (encadré ci-dessus). **Il n'est pas nécessaire d'avoir fait l'exercice précédent pour faire celui-ci et réciproquement**.

(a) Soit $\mathcal{A}$ un attaquant polynomial probabiliste contre CDH avec succès ϵ. Construire un attaquant $\mathcal{B}$ polynomial probabiliste contre $\mathsf{List-CDH}$. Quel est sa probabilité de succès ?

Dans la suite de l'exercice, on considère un attaquant $\mathcal{B}$ polynomial probabiliste contre $\mathsf{List-CDH}$ avec succès ϵ et on veut construire un attaquant $\mathcal{A}$ polynomial probabiliste contre CDH.

(b) Quel est le succès de $\mathcal{A}$ si l'on se contente de retourner un élément pris au hasard dans la liste retournée par $\mathcal{B}$? Qu'en pensez vous ?

On veut construire un attaquant $\mathcal{A}$ ayant une meilleure probabilité de succès. On note dans la suite X, Y deux éléments uniformément distribués dans G un groupe cyclique d'ordre q premier engendré par g. Soit $x \in \mathbf{Z}/q\mathbf{Z}$ tel que $X = g^x$. On choisit s_1 et s_2 uniformément dans $\mathbf{Z}/q\mathbf{Z}$.

(c) Montrer que $X' := g^{s_1}/X^{s_2}$ est uniformément distribué dans G et indépendant de X.

(d) On pose $x' := s_1 - xs_2$ dans $\mathbf{Z}/q\mathbf{Z}$. On considère fixées les valeurs de X, X' et Y. Soit Z et Z' deux autres éléments fixes de G. Montrer qu'avec probabilité au moins $1 - \frac{1}{q}$,

$$Z = Y^x \text{ et } Z' = Y^{x'} \iff Z^{s_2} Z' = Y^{s_1}$$

(e) En déduire une construction de $\mathcal{A}$ en faisant deux appels à $\mathcal{B}$ obtenant un meilleur succès qu'en (b). Que peut on conclure sur les problèmes CDH et $\mathsf{List-CDH}$?

3 Échange de clef et schéma de chiffrement

On définit un protocole d'échange de clef $\mathcal{P}$ entre Alice et Bob à deux passes et sa sécurité. Soit k un paramètre de sécurité, on suppose connu par tous un groupe cyclique G, son ordre q avec $|q| = k$, et un générateur g. On suppose que toutes les quantités échangées et la clef secrète établie sont des éléments de G. Le protocole se déroule ainsi :

1. Bob à partir de G, q, g produit un état s_B et un élément $X \in G$ qu'il envoie à Alice ;
2. Alice à partir de G, q, g produit un état s_A et un élément $Y \in G$ qu'elle envoie à Bob ;
3. Alice calcule à partir de s_A et X une clef $K_A \in G$. De même Bob calcule à partir de s_B et Y une clef $K_B \in G$.

Le protocole $\mathcal{P}$ est correct si $K_A = K_B =: K$. Il est sûr si un adversaire $\mathcal{A}$ observant les données échangées par Alice et Bob ne peut distinguer la clef K établie d'un élément de G aléatoire. Plus formellement on définit $\mathbf{Exp}_{\mathcal{P},k}(\mathcal{A})$:

1. Sous l'entrée 1^k Alice et Bob exécute le protocole $\mathcal{P}$. Ceci produit les quantités échangées X et Y et la clef K éléments de G d'ordre q avec $|q| = k$;
2. on choisit un bit aléatoire $b^\star \xleftarrow{\$} \{0,1\}$. Si $b^\star = 1$ alors $Z := K$ sinon Z est tiré uniformément dans G ;
3. on donne (G, q, g, X, Y, Z) à $\mathcal{A}$ qui sort un bit b ;
4. la sortie de l'expérience est 1 si $b = b^\star$ et 0 sinon.

L'avantage de l'attaquant $\mathcal{A}$ est défini par

$$\mathbf{Adv}_{\mathcal{P},k}(\mathcal{A}) = \left| \Pr\Big(\mathbf{Exp}_{\mathcal{P},k}(\mathcal{A}) = 1\Big) - \frac{1}{2} \right|.$$

Le protocole $\mathcal{P}$ est sûr si pour tout algorithme polynomial probabiliste $\mathcal{A}$ l'avantage $\mathbf{Adv}_{\mathcal{P},k}(\mathcal{A})$ est négligeable.

(a) Que sont $s_B, s_A, X, Y, K_A, K_B, K$ dans le cas du protocole de Diffie-Hellman ? Quelle est l'hypothèse qui assure que le protocole est sûr ?

(b) Montrer qu'à partir de n'importe quel protocole d'échange de clef $\mathcal{P}$ à deux passes, on peut construire un schéma de chiffrement à clef publique Π. Montrer que si $\mathcal{P}$ est sûr alors Π est sémantiquement sûr pour des attaques à clairs choisis.

Partie G. Zémor

4 Soit G un groupe cyclique d'ordre premier q et g_1, g_2 deux générateurs de G. On considère le langage L défini par

$$(g_1, g_2, h_1, h_2) \in L \Longleftrightarrow \exists w \in \mathbf{Z}/q\mathbf{Z} \text{ tel que } \log_{g_1} h_1 = w = \log_{g_2} h_2$$

On considère le protocole suivant destiné à prouver l'appartenance de (g_1, g_2, h_1, h_2) à L :

- Le prouveur choisit r aléatoirement et uniformément dans $\mathbf{Z}/q\mathbf{Z}$ et donne $a_1 = g_1^r$ et $a_2 = g_2^r$ au vérificateur.
- Le vérificateur choisit aléatoirement et uniformément un élément $c \in \mathbf{Z}/q\mathbf{Z}$ et le donne au prouveur.
- Le prouveur calcule $z = r + wc$ modulo q et l'envoie au vérificateur. Ce dernier accepte la preuve si et seulement si $g_1^z = a_1 h_1^c$ et $g_2^z = a_2 h_2^c$.

(a) Démontrer que le protocole est complet, c'est-à-dire que si (g_1, g_2, h_1, h_2) est dans L, le vérificateur accepte la preuve.

(b) Démontrer que le protocole est valide.

(c) Démontrer que le protocole est sans divulgation.

5 Soit $n = pq$ un entier RSA, e un exposant de chiffrement, d l'exposant secret de déchiffrement associé. Le protocole suivant a pour but de prouver au vérificateur V que le prouveur P connaît l'exposant secret $d \mod \phi(n)$.

- V choisit un entier x aléatoire et le donne à P
- P calcule $y = x^d \mod n$ et le donne à V
- V vérifie que $y^e = x \mod n$.

Ce protocole est-il sans divulgation ? Pourquoi ?

6 Soit le langage L constitué des quintuplets (p, q, g, h, k) où p, q, g, h, k sont des entiers vérifiant les propriétés suivantes :

- p est premier
- q est premier et divise $p - 1$
- $g \neq 1$ et $g^q = 1 \mod p$
- il existe deux entiers x et y tels que $xy = 1 \mod q$ et tels que $h = g^x \mod p$ et $k = g^y \mod p$.

On admettra que l'on dispose d'un algorithme polynomial pour vérifier la primalité. Proposer un protocole sans divulgation qui démontre l'appartenance à L. Démontrer successivement que votre protocole est complet, valide, sans divulgation.