République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de Batna 2
Faculté des mathématiques
et de l'informatique
Département d'informatique

Thèse

En vue de l'obtention du diplôme de
Doctorat en Informatique

Méthodes de chiffrement basées sur la factorisation en entiers et logarithme discret

Présentée Par

DJEBAILI Karima

Soutenue le: 16/ 02 / 2017

Membres du jury :

Président:	NOUI Lemnour	Professeur	Université de Batna
Rapporteur:	MELKEMI Lamine	Professeur	Université de Batna
Examinateurs:	SEGHIR Rachid	MCA	Université de Batna
	MOKHTARI Zouhir	MCA	Université de Biskra
	MELKEMI Kamal Eddine	MCA	Université de Biskra

-Á ceux que j'aime.

Remerciements

Grâce à Allah vers lequel vont toutes les louanges, ce travail s'est accompli.

Avant tout, j'adresse mes sincères remerciements à mon directeur de thèse le professeur MELKEMI Lamine pour son soutien tout au long de ces années de thèse. Je ne saurais dire combien nos échanges et ses nombreux conseils m'ont été précieux. Sa disponibilité, ses qualités pédagogiques et humaines, ses compétences et sa bonne humeur m'ont apporté un encadrement déterminant dans toutes les phases de ce travail.

Je remercie le chef de parcours NOUI Lemnouar pour avoir accepté de présider le jury de ma thèse, et aussi pour avoir su me guider avec attention et une gentillesse constante pendant mon parcours. Sa qualité scientifique et humaine, son encouragement et ses remarques ont largement contribué à l'aboutissement de cette thèse.

je tiens à remercier aussi les membres du jury, Mr SEGHIR Rachid, Mr MELKEMI Kamal, et Mr MOKHTARI Zouhir, pour l'honneur qu'ils m'ont fait en acceptant de juger ce travail.

Je souhaite témoigner ma gratitude et vifs remerciements envers tous ceux qui ont contribué, de prés ou de loin à la réalisation de ce travail. Bien sûr, je ne peux terminer

sans remercier mes proches de tout cœur et notamment mes parents qui, au cours de ces années de thèse, m'ont toujours soutenu et encouragé.

Résumé

Dans cette thèse, nous nous sommes intéressés à la sécurité des cryptosystèmes basés sur le problème de la factorisation en entier et le problème du logarithme discret. Nous avons proposé deux nouveaux schémas ainsi qu'une étude efficace de leurs sécurités. Notre premier schéma propose un cryptosystème à clé publique et sa version de signature, qui sont différents de ceux connus dans la littérature. Notre proposition repose sur l'hypothèse raisonnablement admise de sorte qu'il est difficile de trouver l'ordre d'un élément du groupe chaque fois que l'ordre est complètement caractérisé par une équation "difficile à résoudre". En outre, les schémas proposés (chiffrement et signature) ; par opposition au cryptosystème RSA (chiffrement et signature), sont de nature probabiliste et non pas seulement déterministe. Ce schéma de chiffrement est prouvé être IND-CPA (sécurité sémantique). Alors que le second schéma c'est une nouvelle variante du cryptosystème ElGamal qui est sécurisée contre tout adversaire passif et actif. Sous la difficulté de l'hypothèse décisionnelle de Diffie-Hellman, nous pouvons prouver que le schéma proposé est sécurisé contre les attaques adaptatives à texte chiffré choisi. Cette sécurité vérifie non seulement la confidentialité mais aussi vérifie l'authentification et l'intégrité des communications. Le schéma proposé atteint

l'anonymat ainsi que la propriété de robustesse forte.

Mots clés : Problème de la factorisation, problème du logarithme discret, sécurité sémantique, attaque adaptative à texte chiffré choisi.

Abstract

In this thesis, we are interested in the security of cryptosystems based on the factoring problem and the discrete logarithm problem. Our first scheme provides a public key cryptosystem, and its signature version, which are different from that known before in the literature. Our proposal relies on the reasonably accepted assumption that it is hard to find the order of a group element whenever that order is completely characterized by a "hard to solve" equation. Moreover, the proposed schemes (encryption and signature); as opposed to the RSA cryptosystem (encryption and signature); are probabilistic in nature and not just deterministic. This encryption scheme is proven to be IND-CPA (semantic security). While the second scheme is a new variant of ElGamal encryption which is secure against every passive and active adversary. Under the hardiness of the decisional Diffie-Hellman assumption, we can prove that the proposed scheme is secure against adaptive chosen ciphertext attacks. Such security verifies not only the confidentiality but also verifies the authentication and integrity of communications. We display that the modified scheme furthermore achieves anonymity as well as strong robustness.

Keywords : Factorization problem, discrete logarithm problem, semantic security, adaptive chosen ciphertext attacks

ملخص

في هذه الأطروحة نحن مهتمون بالقضايا الأمنية المتعلقة بالخوارزميات التي تعتمد على مشكلة تحليل عدد لعوامله الأولية و مشكلة اللوغريتم المنفصل. اقترحنا مخططين جديدين بالإضافة إلى دراسة فعالة لسالمتهم. مخططنا الأول يقدم نظام تشفير المفتاح العام و الذي يعتمد على مشكلة تحليل عدد لعوامله الأولية، و التي تختلف عن تلك المعروفة سابقا. بالإضافة إلى ذلك، فإن الخطة المقترحة(التشفير والتوقيع)؛ خالفا لترميز RSA (التشفير والتوقيع)، هي احتمالية وليست فقط محددة. في حين أن المخطط الثاني هو عبارة عن مخطط معدل لنظام تشفير " الجمل " وهو آمن ضد أي خصم سواء كان فعال أو غير فعال.

كلمات البحث : مشكلة تحليل عدد لعوامله الأولية, مشكلة اللوغريتم المنفصل ,الأمن.

Table des matières

Notations

Les notations standards suivantes serons utilisées tout au long de cette thèse :

1. $\mathcal{O}$: Oracle.
2. Adv : Avantage.
3. $\mathcal{A}$: Adversaire.
4. sk : La clé secrète.
5. pk : La clé publique.
6. $\mathcal{G}(1^{\lambda})$: Algorithme qui génére les clés.
7. $\mathcal{E}_{pk}(m)$: Algorithme de chiffrement.
8. $\mathcal{D}_{sk}(c)$: Algorithme de déchiffrement.
9. Exp : Experience.
10. ∇ : Un symbole spécial retourné par l'algorithme de déchiffrement en cas d'échec.

Introduction générale

La cryptographie[1] est la pratique et l'étude des stratégies d'une communication sécurisée en présence d'un adversaire. En autre terme, la cryptographie est la génération et l'analyse des protocoles qui empêchent des tiers (ou le public) de lire des messages privés ; divers aspects de la sécurité de l'information telle que la confidentialité, l'intégrité, l'authentification et la non-répudiation des données sont au cœur de la cryptographie moderne. La cryptographie moderne [2] est l'intersection des disciplines des mathématiques, de l'informatique et de l'ingénierie électrique. Les applications de la cryptographie [3] incluent les cartes à puce, les mots de passe, et le commerce électronique etc...

La terminologie de base est que *la cryptographie* se réfère à la science et l'art de la conception des cryptosystèmes ; *la cryptanalyse* est la science et l'art de casser ces cryptosystèmes ; tandis que la *cryptologie*, est l'étude des deux (c.-à-d. la cryptographie et la cryptanalyse). L'entrée d'un processus de chiffrement est communément s'appele le *plaintext* (le texte en clair), et la sortie c'est le *ciphertext* (le texte chiffré). La cryptographie se divise en deux catégories : *Chiffrement symétrique* avoir une clé pour le chiffrement et le déchiffrement, dans ce cas il s'appele chiffrement à clé secrète (ou

privée), ou avoir des clés distinctes pour le chiffrement et le déchiffrement, dans ce cas il s'appele *chiffrement asymétrique* (ou chiffrement à clé publique). Un schéma *de signature numérique* [4] ; un type spécial de primitives de la cryptographique asymétrique ; c'est un système mathématique pour démontrer l'authenticité d'un message ou des documents numériques.

En réalité, le chiffrement asymétrique se base sur la difficulté des fonctions mathématiques qu'est liée à certains factorisations des entiers, logarithme discret, et les relations des courbes elliptiques, même si on peut facilement montrer qu'elles sont inversibles mais elles restent toujours difficiles à inverser. En fait, même avec les connaissances actuelles elles ne puvent pas être inversées en un temps polynomial. Ainsi, la clé publique peut être publiée sans compromettre la sécurité, cette sécurité ne dépend que de garder la clé privée secrète. Les algorithmes à clé publique, contrairement aux algorithmes à clés symétriques, ne nécessitent pas un canal sécurisé pour l'échange initial d'une (ou plusieurs) clé secrète entre les parties.

La sécurité d'un système de chiffrement (soit symétrique soit asymétrique) est censé à refléter l'incapacité d'un adversaire ; étant donné un ciphertext (et toute information publique, comme la clé publique) ; à obtenir des informations sur le plaintext correspondant. La résistance des schémas de chiffrement s'évalue dans différents scénarios d'attaque et chaque scénario est défini en donnant à l'attaquant un but, c.-à-d. ce qu'il va chercher à mettre en défaut dans le système, et des ressources, c.-à-d. les informations auxquelles il aura accès pour réaliser son attaque. Le type de base d'une attaque est *une attaque de texte en clair choisi,* dans lequel l'adversaire peut obtenir des ciphertexts à des messages de son choix. La forte attaque est celle de *l'attaque adaptative à*

texte chiffré choisi [5], dans lequel nous désirons que la confidentialité des données soit maintenue même si l'adversaire a un certain accès (limité) à un "oracle de décryptage", ce qu'est une boîte qui contient la clé secrète de déchiffrement et met en œuvre le déchiffrement sous cette clé.

L'objectif de cette thèse est lié justement à la sécurité de deux grands systèmes dans le chiffrement asymétrique ; le cryptosystème RSA [6] et le cryptosystème ElGamal [7] ; Nous proposons deux nouvelles méthodes pour forcer la sécurité de ces derniers. Dans la première méthode, nous introduisons un nouveau problème appelé le problème de trouver l'ordre d'un élément dans un groupe (*Order Finding Problem* (OFP)), nous proposons, à partir de ce problème, deux nouveaux systèmes le premier pour le chiffrement et le second pour la signature, la sécurité de cette méthode repose sur le fait que ce problème devient impossible à résoudre calculatoirement pour n un produit de deux nombres premiers suffisamment grands. Dans la deuxième méthode nous proposons ce qu'on s'appelle un schéma de chiffrement ElGamal modifié (*Modified ElGamal Encryption Scheme* (MEGES)), et prouvons qu'il est sécurisé contre une attaque adaptative à texte chiffré choisi, aussi nous prouvons que le MEGES atteint l'anonymat ainsi que la propriété de robustesse forte. Cette thèse est organisée en cinq chapitres en plus d'une introduction et d'une conclusion générales :

Le premier chapitre identifie les types d'attaques sur des primitives et des protocoles cryptographiques. Nous abordons d'abord les problèmes mathématiques difficiles à résoudre où de nombreux cryptosystèmes à clé publique ont été proposés depuis ils sont le plus souvent basés sur ces problèmes. Ensuite nous présentons les notions de sécurité pour le chiffrement à clé publique afin de choisir le niveau de sécurité

adéquat. Après cela nous détallions la sécurité computationnelle qui mesure la quantité d'effort de calcul nécessaire par les meilleures méthodes actuellement connues, pour vaincre un cryptosystème, à la fin nous montrons les séparations et les implications entre les principales notions de sécurité dans le cadre du chiffrement asymétrique.

Le but de second chapitre est de présenter brièvement quelques cryptosystèmes dont la sécurité est fondée sur des hypothèses du calcul liées au problème de la factorisation des entiers. En particulier, nous étudions le cryptosystème RSA, le cryptosystème de Goldwasser-Micali et le cryptosystème de Paillier. Nous présentons également leurs sécurités.

Au-delà de la factorisation des entiers, un autre problème du calcul d'une grande importance cryptographique est celui du logarithme discret dans certains groupes finis, tels que $\mathbb{Z}_p^*$. Plusieurs systèmes cryptographiques ont été reportés à ce problème, nous nous concentrons dans le troisième chapitre sur le cryptosystème Elgamal et ses variantes.

La dernière partie présente nos travaux, cete partie est divisée en deux chapitres : le premier est consacré à notre nouveau cryptosystème probabiliste à clé publique et sa signature qui se basent sur le problème de la factorisation des entiers, et aussi consacrée à une étude détaillée de la sécurité. Dans le deuxième chapitre, nous proposons notre deuxième contribution, ce qui est une variante sécurisée et robuste du cryptosystème ElGamal. Les comparaisons avec les meilleures solutions, proposées dans la littérature, montrent que notre proposition offre un niveau de sécurité élevé.

Enfin, dans la partie conclusion générale, nos contributions sont résumées.

L’état de l’art

Chapitre 1

Les notions de sécurité

1.1 Introduction

Les schèmes de la cryptographie à clé publique se reposent souvent sur des algorithmes cryptographiques basés sur des problèmes mathématiques qui n'admettent actuellement aucune solution efficace.

Le but de ce chapitre est de présenter certains problèmes difficiles et de mettre en évidence les relations entre les notions de sécurité des schèmes basés sur ces problèmes.

1.2 Préliminaires

Avant d'aborder les problèmes difficiles nous avons besoin de définir ce qu'est une fonction à sens unique. Cela dépend de la difficulté de la calculer, nous devons donc introduire quelques notions :

- *Une fonction à sens unique* est une fonction bijective facile à calculer, mais sa fonction inverse est censée être difficile à calculer.

- *Un adversaire* est modélisé comme un algorithme probabiliste et polynomial, ou dans certains cas un ensemble de tels algorithmes. Nous utilisons la notation $\mathscr{A}$ pour un adversaire.

- *Une expérience* est un jeu qui retourne 0 ou 1. Il est composé de calculs polynomiaux et une occurrence de l'adversaire $\mathscr{A}$.

- *Un avantage* est une mesure probabiliste de la difficulté d'une expérience donnée. Un problème donné est difficile si l'avantage de l'expérience est négligeable. Nous le notons par Adv.

- *Le paramètre de sécurité* est une variable qui mesure la taille d'entrée du problème du calcul. Les besoins en ressources de l'algorithme cryptographique ou de protocole, ainsi que la probabilité de l'adversaire pour casser un algorithme, sont exprimés en termes de ce paramètre. Le paramètre de sécurité λ est communément exprimé en représentation unaire 1^λ (ce qui signifie une chaîne de longueur λ de 1)

- *Schéma de chiffrement à clé publique* (noté par $\Omega = (\mathscr{G}, \mathscr{E}, \mathscr{D})$) comprend les trois algorithmes suivants :

  – *Génération de clés :* un algorithme polynomial qui prend un paramètre de sécurité 1^λ en entrée et retourne une paire de clés (pk, sk), pour simplifier, nous écrivons $(pk, sk) \leftarrow \mathscr{G}(1^\lambda)$. Chaque système de chiffrement à clé publique est associé à un espace de messages en clair $MsgSp_{(pk,sk)}$ et un espace de messages chiffrés $CiphSp_{(pk,sk)}$.

  – *Chiffrement :* un algorithme polynomial qui prend le pair (pk, m) en entrée, où $m \in MsgSp_{(pk,sk)}$ et renvoie un chipertext $c \in CiphSp_{(pk,sk)}$, pour simplifier, nous écrivons $c \leftarrow \mathscr{E}_{pk}(m)$.

  – *Déchiffrement :* un algorithme polynomial qui prend le pair (sk, c) en entrée, où $c \in CiphSp_{(pk,sk)}$ et renvoie un chipertext $m \in MsgSp_{(pk,sk)}$ ou le symbole ∇ si $m \notin MsgSp_{(pk,sk)}$, pour simplifier, nous écrivons $m \leftarrow \mathscr{D}_{sk}(c)$.

1.3 Les problèmes difficiles

Maintenant, nous pouvons expliquer certains problèmes difficiles. Tout d'abord, nous rappelons brièvement le problème de la factorisation en entiers et le problème du logarithme discret. Nous nous concentrons par la suite sur le problème de Diffie-Hellman qui peut être un problème de calcul ou de décision.

1.3.1 Le problème de la factorisation

Certains algorithmes cryptographiques sont conçus autour des hypothèses de difficulté de factoriser un grand nombre, ce qui rend ces algorithmes difficiles à casser dans la pratique par un adversaire. Il est théoriquement possible de casser un tel algorithme, mais il est impossible de le faire par tous les moyens pratiques connus.

Définition 1.3.1 *Nous définissons un générateur de paramètres de la factorisation des entiers, comme un algorithme polynomial FCgen qui prend en entrée un paramètre de sécurité* 1^λ*, et génère un entier* $n = pq$.

Définition 1.3.2 *Le problème de la factorisation (FC) d'un entier est le suivant : étant donné un entier positif n, trouver sa décomposition en facteurs premiers ; c.-à-d. écrire* $n = p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$*, où les* p_i *sont premiers entre eux deux à deux et chaque* $e_i \geq 1$.

Hypothès 1.3.1 . *Considérer l'expérience suivante associée à un adversaire* $\mathscr{A}$.

$Exp^{FC}_{\mathscr{A},FCgen}(1^\lambda)$

$(p,q) \leftarrow FCgen(1^\lambda)$

$n \leftarrow pq$

$p', q' \leftarrow \mathscr{A}(1^\lambda, n)$

Si $n = p'q'$ *retourner 1 sinon retourner 0.*

L'avantage de $\mathscr{A}$ *dans l'expérience ci-dessus est défini comme :*

$$Adv^{FC}_{\mathscr{A}}(\lambda) = |Pr[Exp^{FC}_{\mathscr{A},FCgen}(1^\lambda) = 1]|. \tag{1.1}$$

L'hypothèse de FC exprime que $Adv^{FC}_{\mathscr{A}}(\lambda)$ est négligeable pour tout $\mathscr{A}$.

Le cryptosystème le plus célèbre basé sur la factorisation est probablement RSA, nous définissons le problème RSA comme suit :

Définition 1.3.3 *Le problème RSA (PRSA) est le suivant : étant donné un entier positif n qui est un produit de deux nombres premiers p et q , impairs et distincts, un entier positif e tel que* $PGCD(e, \phi(n)) = 1$ *et un entier c, trouver un entier m tel que* $m^e \equiv c \mod n$.

En d'autres termes, le problème RSA est celui de trouver le e^{th} racines modulo un entier composé n.

Hypothès 1.3.2 . *Considérer l'expérience suivante associée à un adversaire* $\mathscr{A}$.

$Exp^{PRSA}_{\mathscr{A},FCgen}(1^\lambda)$

$(p, q) \leftarrow FCgen(1^\lambda)$

$n \leftarrow pq$

$e, c \leftarrow \mathbb{Z}^*_n$

$m' \leftarrow \mathscr{A}(1^\lambda, n)$

Si $m' = c^e \mod n$ *retourner 1 sinon retourner 0.*

L'avantage de $\mathscr{A}$ dans l'expérience ci-dessus est défini comme :

$$Adv_{\mathscr{A}}^{PRSA}(\lambda) = |Pr[Exp_{\mathscr{A},FCgen}^{PRSA}(1^{\lambda}) = 1]|. \tag{1.2}$$

L'hypothèse de PRSA exprime que $Adv_{\mathscr{A}}^{PRSA}(\lambda)$ est négligeable pour tout $\mathscr{A}$.

Corollaire 1.3.1 *PRSA ≤ FC. Autrement dit, le problème RSA est réduit au problème de la factorisation.*

Il est largement admis que le RSA et le problème de la factorisation sont calculatoirement équivalents, bien qu'aucune preuve soit connue.

1.3.2 Le problème du logarithme discrete

La sécurité des schémas cryptographiques à clé publique est souvent ramenée à la difficulté de résolution du logarithme discret (DL) [8], ce dernier est défini comme suit :

soit G un groupe cyclique généré par un générateur g. Étant donné $h \in G$, trouver le logarithme discret de h à la base g. L'hypothèse du DL indique qu'il n'existe pas un algorithme polynomial qui permet de résoudre ce problème.

Définition 1.3.4 *Nous définissons un générateur de paramètres du logarithme discrète comme un algorithme du temps polynomial DLgen qui prend en entrée un paramètre de sécurité 1^{λ}, et génère un entier premier p avec la description d'un groupe cyclique G d'ordre p.*

Hypothès 1.3.3 . *Considérer l'expérience suivante associée à un adversaire $\mathscr{A}$.*

$Exp^{DL}_{\mathscr{A},DLgen}(1^\lambda)$

$(G,g,p) \leftarrow DLgen(1^\lambda)$

$a \leftarrow \mathbb{Z}_p$

$a' \leftarrow \mathscr{A}(1^\lambda, G, p, g, g^a)$

Si $a = a'$ retourner 1 sinon retourner 0.

L'avantage de $\mathscr{A}$ dans l'expérience ci-dessus est défini comme :

$$Adv^{DL}_{\mathscr{A}}(\lambda) = |Pr[Exp^{DL}_{\mathscr{A},DLgen}(1^\lambda) = 1]|. \tag{1.3}$$

L'hypothèse de DL exprime que $Adv^{DL}_{\mathscr{A}}(\lambda)$ est négligeable pour tout $\mathscr{A}$.

Quelques exemples de groupes dans lesquels le problème du DL est censé être difficile sont : $\mathbb{Z}^*_p$ pour un grand nombre premier p où $p-1$ a au moins un facteur premier grand, le sous-groupes cyclique $H \subset \mathbb{Z}_p$ d'ordre premier q et certains groupes de courbes elliptiques.

1.3.3 Le problème Diffie-Hellman

Nous commençons par présenter le problème de Diffie-Hellman. Puis nous considérons deux situations. La première le problème calculatoire de Diffie-Hellman (CDH) qui consiste à produire le correct résultat et la seconde le problème décisionnel de Diffie –Hellman qu'est utilisé pour déterminer si une valeur donnée est correcte.

Soit g un générateur d'un groupe cyclique d'ordre premier p.

$$A \rightarrow B : g^a$$

$$B \rightarrow A : g^b$$

$$A \rightarrow B : g^{ab}.$$

Le problème calculatoire de Diffie-Hellman (CDH)

Le problème CDH [9], est défini comme suit : étant donné une instance aléatoire (g, g^a, g^b) où $a, b \in G$, calculer g^{ab}. L'hypothèse CDH affirme qu'il n'existe pas un algorithme polynomial qui permet de résoudre ce problème.

Hypothès 1.3.4 *Considérer l'expérience suivante associée à un adversaire $\mathscr{A}$.*

$Exp^{CDH}_{\mathscr{A},DLgen}(1^\lambda)$

$(G, g, p) \leftarrow DLgen(1^\lambda)$

$a, b \leftarrow \mathbb{Z}_p$

$Z \leftarrow \mathscr{A}(1^\lambda, G, p, g, g^a, g^b)$

Si $Z = g^{ab}$ retourner 1 sinon retourner 0.

L'avantage de $\mathscr{A}$ dans l'expérience ci-dessus est défini comme :

$$Adv^{CDH}_{\mathscr{A}}(\lambda) = |Pr[Exp^{CDH}_{\mathscr{A},DLgen}(1^\lambda) = 1]|. \tag{1.4}$$

L'hypothèse de CDH exprime que $Adv^{CDH}_{\mathscr{A}}(\lambda)$ est négligeable pour tout $\mathscr{A}$.

Le problème décisionnel de Diffie-Hellman (DDH)

Le problème DDH [10], déclare que étant donné deux distributions (g, g^a, g^b, g^{ab}), (g, g^a, g^b, g^c) où $a, b, c \in G$, il est difficile de distinguer ces deux distributions. Cela signifie qu'obtenir des informations à propos de g^{ab} en donnant (g, g^a, g^b) est difficile.

Hypothès 1.3.5 *Considérer l'expérience suivante associée à un adversaire* $\mathscr{A}$.

$Exp^{DDH,\beta}_{\mathscr{A},DLgen}(1^\lambda)$

$(G,g,p) \leftarrow DLgen(1^\lambda)$

$a,b,c \leftarrow \mathbb{Z}_p$

Si $\beta = 1$ *alors* $T = g^{ab}$ *sinon* $T = g^c$.

$\beta' \leftarrow \mathscr{A}(1^\lambda, G, p, g, g^a, g^b, T)$

Retourner β'

L'avantage de $\mathscr{A}$ *dans l'expérience ci-dessus est défini comme :*

$$Adv^{DDH}_{\mathscr{A}}(\lambda) = |Pr[Exp^{DDH,\beta}_{\mathscr{A},DLgen}(1^\lambda) = \beta : \beta \leftarrow \{0,1\}] - \frac{1}{2}|. \tag{1.5}$$

qui peut également être écrit comme :

$$Adv^{DDH}_{\mathscr{A}}(\lambda) = |Pr[Exp^{DDH,1}_{\mathscr{A},DLgen}(1^\lambda) = 1] - Pr[Exp^{DDH,0}_{\mathscr{A},DLgen}(1^\lambda) = 1]|. \tag{1.6}$$

L'hypothèse de DDH exprime que $Adv^{DDH}_{\mathscr{A}}(\lambda)$ est négligeable pour tout $\mathscr{A}$.

S'il existe un algorithme polynomial qui peut résoudre le DL il pourra trivialement résoudre le problème CDH. Autrement dit, si a et b peuvent être dérivés à partir de (g^a, g^b) il devient facile à calculer g^{ab}. Par conséquent, dans un groupe où l'hypothèse CDH est difficile implique immédiatement que l'hypothèse DL est difficile aussi. Aucune preuve mathématique ne démontre la relation inverse.

D'autre part, si un algorithme puissant pourrait résoudre CDH, c-à-d., dérivée g^{ab} à partir de (g^a, g^b) seulement, il serait devenu trivial de distinguer (g, g^a, g^b, g^{ab}) et (g, g^a, g^b, g^c). Encore une fois, la relation inverse ne peut pas être prouvée.
Par conséquent, la relation entre le DL, le CDH et le DDH est souvent écrit comme suit :

$$DDH \Rightarrow CDH \Rightarrow DL. \tag{1.7}$$

La notation "⇒" est traduite en "implique immédiatement". (Voir [11], qui présente un aperçu des relations entre DL, CDH et DDH).
Dans un groupe où DDH est difficile le CDH et le DL seront difficiles également. Au contraire, il existe des groupes où le CDH et l'hypothèse de DL sont difficiles, mais le DDH peut être résolu facilement. Tel groupes sont appelés groupes *Diffie-Hellman de gap*[12], ou simplement des groupes de *gap*.

1.4 Les notions d'attaques

A ce stade, nous avons besoin d'introduire divers modèles d'attaque. Il existe trois modèles d'attaque de base sont classées par ordres croissants de la difficulté :

- **Attaque passive** : parfois appelée une attaque à texte clair choisi, souvent notée CPA.
- **Attaque à texte chiffré choisi** : souvent notée CCA1.
- **Attaque adaptative à texte chiffré choisi** : souvent notée CCA2 .

1.4.1 L'attaque passive

Une attaque passive est une forme très faible d'attaque. L'adversaire est autorisé à examiner divers messages cryptés. Il a également un accès à un oracle, qui effectue le cryptage, mais pas le décryptage. Par conséquent, ce modèle d'attaque est une attaque simple sur un système à clé publique, puisque dans ce système tout le monde, y compris l'attaquant, a un accès de la fonction de cryptage.

1.4.2 L'attaque à texte chiffré choisi

Une attaque à texte chiffré choisi (CCA1) [13], représente une forme légèrement plus forte d'attaque. L'adversaire dans ce cas a un accès à un oracle qui effectue le décryptage. L'adversaire peut demander à l'oracle de décrypter un certain nombre polynomial de textes chiffrés de son choix. Après cela, il reçoit un texte chiffré cible et a demandé à le décrypter, ou trouver des informations sur le texte en clair correspondant, sans l'aide de l'oracle.

1.4.3 L'attaque adaptative à texte chiffré choisi

une attaque adaptative à texte chiffré choisi (CCA2) [5], est une forme très forte d'attaque, l'adversaire est désormais autorisé à demander à l'oracle de décryptage pour décrypter tout ciphertext de son choix, sauf le ciphertext cible. Il est largement considéré que tout nouvel algorithme proposé de chiffrement à clé publique doit répondre à l'exigence de la sécurité contre une attaque adaptative à texte chiffré choisi.

1.4.4 Résumé d'attaques

- Pour l'attaque CPA, $\mathcal{O}_1 = \mathcal{O}_2 = \emptyset$, l'adversaire n'a pas un accès aux oracles.
- Pour l'attaque CCA1, $\mathcal{O}_1 = \{\mathcal{D}\}$,$\mathcal{O}_2 = \emptyset$, l'adversaire peut utiliser l'oracle de décryptage avant de recevoir le texte chiffré cible c.
- Pour CCA2, $\mathcal{O}_1 = \mathcal{O}_2 = \{\mathcal{D}\} - c$, l'adversaire peut utiliser l'oracle de décryptage avant et après de recevoir le texte chiffré cible c.

où $\mathcal{O}_1$ et $\mathcal{O}_1$ sont des oracles de décryptage mis à la disposition de l'adversaire après et avant de recevoir le texte chiffré respectivement.

Plus l'attaquant a des accès aux différents oracles, plus il est puissant. Cette affirmation évidente peut être mathématiquement traduite à :

$$CCA2 \Rightarrow CCA1 \Rightarrow CPA.$$

1.5 Les notions de sécurité pour le chiffrement à clé publique

Après les notions d'attaques, nous nous concentrons maintenant sur les notions de sécurité. Nous insistons plus précisément sur trois notions qui représentent trois niveaux de sécurité. Nous détaillons ces notions de la plus faible à la plus forte.

1.5.1 La fonction à sens unique

une fonction de chiffrement est dite être à sens unique (ou one-way function en anglais (OW)) [14], si pour tout adversaire $\mathscr{A}$, avec l'exécution d'un algorithme probabiliste et polynomial, il ne peut pas récupérer le plaintext m à partir du ciphertext c avec un avantage non négligeable.

Définition 1.5.1 *nous disons qu'un système de chiffrement fournit une fonction à sens unique si l'avantage suivant est négligeable :*

$$Adv_{\mathscr{A}} = Pr[m \leftarrow \mathscr{A}(pk, c) | (pk, sk) \leftarrow \mathscr{G}(1^{\lambda}), c \leftarrow \mathscr{E}_{pk}(m)]. \tag{1.8}$$

1.5.2 La sécurité sémantique

Un système de chiffrement à clé publique est sémantiquement sécurisé [15], si et seulement si le ciphertext ne révèle aucune information partielle, que ce soit sur le plaintext, qui puisse être calculé dans un temps polynomial. Parfois appelée *indistinguabilité* (IND).

Définition 1.5.2 *Soit $\mathscr{A} = (\mathscr{A}_1, \mathscr{A}_2)$ un adversaire, considérer le scénario suivant :*

1. *$\mathscr{A}_1$ a donné la clé publique.*
2. *$\mathscr{A}_1$ choisit deux messages m_0 et m_1.*
3. *$b \in \{0,1\}$ est choisi au hasard et $c = \mathscr{E}_{pk}(m_b)$ a donné à $\mathscr{A}_2$.*
4. *$\mathscr{A}_2$ sortie b'.*

La sécurité sémantique signifie que la probabilité suivante est négligeable :

$$Pr[b = b'] - \frac{1}{2}. \tag{1.9}$$

Même si la sécurité sémantique implique plus de sécurité que le OW, il y a encore quelques manques de sécurité parce qu'il est encore possible de modifier le ciphertext. Cette modification peut permettre à l'adversaire de produire un nouveau ciphertext où le texte décrypté est associé au plaintext initial.

1.5.3 La non malléabilité

La non-malléabilité (NM) [16], est la notion la plus forte de la sécurité. Il correspond à faire passer le message dans une boîte noire avec aucune possibilité de toucher le message. L'adversaire n'a maintenant aucune possibilité de faire n'importe quel calcul sur ce dernier. Plus formellement, l'adversaire ne devrait pas être en mesure de produire un nouveau ciphertext de telle sorte que les textes en clair sont significativement reliés.

Définition 1.5.3 *Soit $\mathscr{A} = (\mathscr{A}_1, \mathscr{A}_2)$ un adversaire, considérer le scénario suivant :*

1. *$\mathscr{A}_1$ a donné la clé publique.*
2. *$\mathscr{A}_1$ choisit un espace de message M.*
3. *Deux messages m et m' sont choisis au hasard dans M et $c = \mathscr{E}_{pk}(m)$ a donné à $\mathscr{A}_2$.*
4. *$(R, y) \leftarrow \mathscr{A}_2(M, c)$.*

La non-malleabilitéb signifie que la probabilité suivante est négligeable :

$$Pr[R(m,x)] - Pr[R(m',x)]. \tag{1.10}$$

où $x = \mathscr{D}_{sk}(y)$ et R est une relation binaire. NM est le plus fort niveau d'exigences de sécurité. Évidemment, il implique IND ce qui implique OW :

$$NM \Rightarrow IND \Rightarrow OW.$$

1.6 La sécurité computationnelle

La sécurité computationnelle [17] est le lien entre les adversaires et les notions de sécurité. Laissez appeler XXX les adversaires (XXX est CPA, CCA1 ou CCA2). Nous rappelons que :

- Pour CPA, $\mathscr{O}_1 = \mathscr{O}_2 = \emptyset$.
- Pour CCA1, $\mathscr{O}_1 = \{\mathscr{D}\}, \mathscr{O}_2 = \emptyset$.
- Pour CCA2, $\mathscr{O}_1 = \mathscr{O}_2 = \{\mathscr{D}\} - c$.

Un jeu correspond à un scénario d'attaque, pour lequel nous pouvons calculer l'avantage de l'attaquant. Cela donne une sorte de mesure le niveau de sécurité. Le graphe ; dans la prochaine section ; montre l'implication entre une situation et une autre et aussi quel type de sécurité peut être déduit d'une autre situation.

1.6.1 Les jeux IND-XXX

Donnons un système de cryptage $\Omega = (\mathscr{G}, \mathscr{E}, \mathscr{D})$. Un adversaire est une paire $\mathscr{A} = (\mathscr{A}_1, \mathscr{A}_2)$ des algorithmes probabilistes et polynomiaux. Laissez $b \in \{0,1\}$ et $IND^b_{XXX}(\mathscr{A})$ être l'algorithme suivant :

- $(pk, sk) \leftarrow \mathscr{G}(1^\lambda)$.
- $(m_1, m_2) \leftarrow \mathscr{A}_1^{\mathscr{O}_1}(pk)$.
- $b' \leftarrow \mathscr{A}_2^{\mathscr{O}_2}(pk, \mathscr{E}_{pk}(m_b))$.
- retourner b'.

Nous définissons $Adv^{IND_{xxx}}_{\Omega,\mathscr{A}}$:

$$Adv^{IND_{xxx}}_{\Omega,\mathscr{A}} = Pr[b' \leftarrow IND^1_{xxx}(A) : b' = 1] - Pr[b' \leftarrow IND^0_{xxx}(A) : b' = 1]. \quad (1.11)$$

un systéme de chiffrement est IND-XXX sécurisé si pour tout adversaire $\mathscr{A}$ l'$Adv^{IND_{xxx}}_{\Omega,\mathscr{A}}$ est négligeable.

Cet avantage peut aussi être exprimé comme suit :

$$\begin{aligned} Adv^{IND_{xxx}}_{\Omega,\mathscr{A}} &= Pr[b' \leftarrow IND^1_{xxx}(A) : b' = 1] - Pr[b' \leftarrow IND^0_{xxx}(A) : b' = 1] \\ &= 2Pr[b' \leftarrow IND^b_{xxx}(A) : b' = b] - 1. \end{aligned} \quad (1.12)$$

1.6.2 Les jeux NM-XXX

Donnons un système de cryptage $\Omega = (\mathscr{G}, \mathscr{E}, \mathscr{D})$. Un adversaire est une paire $\mathscr{A} = (\mathscr{A}_1, \mathscr{A}_2)$ des algorithmes probabilistes et polynomiaux. Pour $b \in \{0,1\}$, laissez $Exp^{atk-b}_{\Omega,\mathscr{A}}$ être l'expérience suivante :

- $(pk, sk) \leftarrow \mathscr{G}(1^\lambda)$.
- $(M) \leftarrow \mathscr{A}_1^{\mathscr{O}_1}(pk); m_0, m_1 \leftarrow M$.
- $y \leftarrow \mathscr{E}_{pk}(m_b); (\mathscr{R}, \tilde{y}) \leftarrow \mathscr{A}_2^{\mathscr{O}_2}(M, y); \tilde{x} \leftarrow \mathscr{D}(\tilde{y})$.
- retourner 1 si $(y \notin \tilde{y}) \wedge (\nabla \notin \tilde{x}) \wedge \mathscr{R}(m_b, \tilde{x})$ sinon retourner 0.

$$Adv^{NM_{xxx}}_{\Omega,\mathscr{A}} = Pr[Exp^{atk-1}_{\Omega,\mathscr{A}} = 1] - Pr[Exp^{atk-0}_{\Omega,\mathscr{A}} = 1]. \tag{1.13}$$

Un schéma de chiffrement est NM-XXX sécurisé, si pour tout adversaire $\mathscr{A}$ l' $Adv^{IND_{xxx}}_{\Omega,\mathscr{A}}$ est négligeable.

1.7 Les relations entre les notions de sécurité

Pour résumer tous les différents types de jeux, le graphe suivant (Voir figure 1.1), représente les implications et les non-implications entre eux avec leur niveau de sécurité correspondant [18].

Pour être considéré comme sécurisé, un système doit être au moins IND-CPA sécurisé. Ce graphe montre que IND-CCA2 est équivalent à NM-CCA2, il est donc suffisant

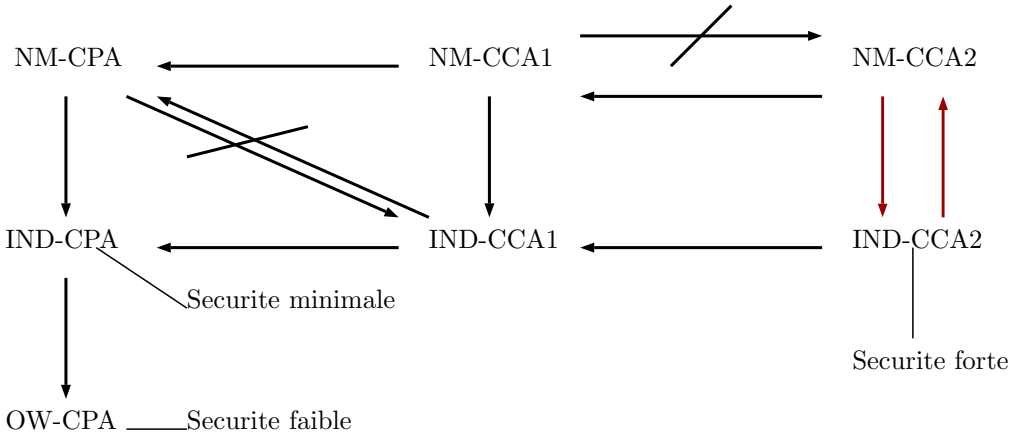

FIGURE 1.1: Relations entre les notions de sécurité pour un schéma de chiffrement à clé publique

pour prouver IND-CCA2 sécurité être sûr d'avoir la sécurité NM-CCA2. Ces deux types de sécurité correspondent à une sécurité très forte.

Deux autres propriétés, qui ont fait l'objet d'études formels dans la sécurité des cryptosystèmes, sont l'anonymat [19] et la robustesse forte [20]. L'anonymat se réfère à la propriété d'un ciphertext qui ne révèle pas la clé publique sous laquelle il a été chiffré, tandis que la forte robustesse fournit une couche de protection contre les abus ou les erreurs en veillant qu'un texte chiffré peut uniquement déchiffrer par l'utilisateur visé.

1.8 L'anonymat et la robustesse forte

L'anonymat (ANON) est définie par l'expérience suivante :

1. $\mathscr{A}$ choisit deux clés publiques pk_0, pk_1 et les envoyer à le challenger.

2. Le challenger choisit $b \in \{0,1\}$, calcule $c_b = \mathscr{E}_{pk_b}(m)$ et le donne à $\mathscr{A}$.

3. $\mathscr{A}$ calcule $b' = \mathscr{A}(pk_0, pk_1, c_b)$.

4. $\mathscr{A}$ réussit si et seulement si $b' = b$.

$$EXP_{\mathscr{A},\Omega}^{ANON} = \begin{cases} 1 & \text{si } b' = b \\ 0 & \text{sinon.} \end{cases}$$

Nous disons qu'un cryptage est anonyme si la probabilité suivante est négligeable :

$$Pr\{EXP_{\mathscr{A},\Omega}^{ANON} = 1\}. \tag{1.14}$$

Nous illustrons le concept de la robustesse forte, (strong robustness en anglais (SROB)), par l'expérience suivante :

1. $\mathscr{A}$ choisit deux clés publiques pk_0, pk_1 et les envoyer à le challenger.

2. Le challenger choisit $b \in \{0,1\}$, calcule $c_b = \mathscr{E}_{pk_b}(m)$ et le donne à $\mathscr{A}$.

3. $\mathscr{A}$ interroge l'oracle de decryptage pour (sk_0, c_b) et pour (sk_1, c_b) puis cet oracle renvoie m_0 et m_1 respectivement.

4. $\mathscr{A}$ réussit si et seulement si $m_1 \neq \nabla$ et $m_2 \neq \nabla$.

$$EXP_{\mathscr{A},\Omega}^{SROB} = \begin{cases} 1 & \text{si } m_1 \neq \nabla \text{ et } m_2 \neq \nabla \\ 0 & \text{sinon.} \end{cases}$$

Nous disons qu'un cryptage est fortement robustesse si la probabilité suivante est négligeable :

$$Pr\{EXP_{\mathscr{A},\Omega}^{SROB} = 1\}. \tag{1.15}$$

1.9 Les modèles de preuves de sécurité en cryptographie

Nombreuses preuves de sécurité sont faites en admettant que tous les participants aient le droit de faire appel à une suite aléatoire (parfaite). On dit que ces preuves sont faites dans le modèle de l'oracle aléatoire. Dans le cas contraire elles sont faites dans le modèle standard.

Le modèle de l'oracle aléatoire

Les oracles aléatoires ont d'abord été utilisés dans les preuves cryptographiques rigoureuses dans la publication de Mihir Bellare et Phillip Rogaway [21]. Ils sont généralement utilisés lorsque les fonctions d'hachage cryptographique, dans un cryptosystème, ne peuvent être prouvées de posséder les propriétés mathématiques requises par la preuve. Un modèle de l'oracle aléatoire c'est un modèle dans lequel on effectue des preuves de sécurité en supposant que toutes les parties qui interviennent ont un accès à un générateur véritablement aléatoire (pas pseudo-aléatoire).

Le modèle standard

les systèmes qui nécessitent seulement une ou plusieurs propriétés ayant une définition dans le modèle standard, (comme la résistance à la collision, la résistance pré-image, la deuxième résistance pré-image, etc.), peuvent souvent être prouvés sécurisés dans le modèle standard (par exemple, le cryptosystème Cramer-Shoup[22]).

Signification d'une preuve dans le modèle de l'oracle aléatoire

De nombreux systèmes ont été prouvés sécurisés dans le modèle de l'oracle aléatoire, par exemple [23], [24] et [25]. En [26], Amos Fiat et Adi Shamir ont montré une application majeure des oracles aléatoires pour la création de signatures.

Russell Impagliazzo et Steven Rudich [27] ont montré la limitation des oracles aléatoires, à savoir que leur seule existence ne suffit pas pour l'échange d'une clé secrète d'une façon confidentielle.

Il est connu qu'une preuve dans le modèle de l'oracle aléatoire n'implique pas que le système est sûr dans la vie pratique [28].

1.10 Conclusion

Les définitions classiques de la sécurité pour les cryptosystèmes ont principalement concerné avec le secret des données cryptées. En particulier, la notion largement acceptée est celle de la non-malléabilité qu'est dirigée à capturer divers aspects de secret des données dans la cryptographie. Cependant, depuis les cryptosystèmes sont utilisés dans une grande gamme d'application, on les oblige souvent à satisfaire des propriétés supplémentaires. Deux de ces propriétés sont l'anonymat et la robustesse forte.

Dans le chapitre suivant, nous rappelons quelques-uns des principaux problèmes difficiles basés sur la factorisation utilisés en cryptographie.

Chapitre 2

Cryptosystèmes à clé publique basé sur la factorisation

2.1 Introduction

La cryptographie est la science de l'utilisation des mathématiques pour chiffrer et déchiffrer les données [2] ces données peuvent être des communications privées, des transactions bancaires par carte de crédit sur le web, e-mail et des mots de passe [29]. La cryptographie vous permet de stocker ces informations sensibles ou de les transmettre à travers des réseaux non sécurisés (comme l'internet) de sorte qu'elles ne peuvent pas être lues par n'importe quelle personne, sauf le destinataire prévu. La solution classique à ce problème est s'appelé chiffrement à clé privée (ou secrète). La solution moderne est le chiffrement à clé publique, ce dernier basé sur une fonction à sens unique [30]. La cryptographie est également utilisée pour une autre variété de sécurité, y compris les signatures électroniques [4], qui sont utilisées pour prouver qui a envoyé un message.

La première catégorie de problèmes cryptographiques va être étudiée dans cette thèse est celle des problèmes basés sur la factorisation en entiers. Factoriser un nombre consiste à le décomposer en produit de facteurs premiers. C'est l'un des problèmes calculatoires qui ont été étudiés depuis le plus longtemps, mais, à ce jour, aucun algorithme polynomial n'est connu pour le résoudre. Cette famille a été utilisée dans [6] pour présenter le révolutionnaire concept de schémas de signature et de chiffrement à clé publique.

Plusieurs cryptosystèmes ont été proposés pour forcer la sécurité. Dans ce chapitre, nous allons passer en revue ces cryptosystèmes et identifier leurs avantages et incon-

vénients en étudiant particulièrement leur sécurité sémantique.

2.2 Le cryptosystème RSA

Le cryptosystème RSA [6], du nom de ses inventeurs R. Rivest, A. Shamir et L. Adleman, est le système de chiffrement à clé publique le plus largement utilisé. Il peut être utilisé pour fournir à la fois la confidentialité (le chiffrement) et l'authentification (la signature numérique), sa sécurité est basée sur le problème RSA.

Le système RSA de base est le suivant :

1. $(pk, sk) \leftarrow \mathscr{G}(1^\lambda)$ **:** Choisir p et q deux nombres premiers distincts, générer la clé publique (n, e) et la clé secrète (n, d), tel que $n = pq$, e coprime avec $\phi(n) = (p-1)(q-1)$ et $d = e^{-1} \bmod \phi(n)$.

2. $c \leftarrow \mathscr{E}_{pk}(m)$ **:** Donner un message $m \in \mathbb{Z}_n^*$, $c = m^e \bmod n$.

3. $m \leftarrow \mathscr{D}_{sk}(c)$ **:** Donner un message chiffré $c \in \mathbb{Z}_n^*$, $m = c^d \bmod n$.

2.2.1 La sécurité du RSA

Casser le cryptosystème RSA est au moins aussi difficile que le problème de la factorisation. Factoriser un grand nombre n'est pas "prouvablement" difficile, mais aucun algorithme n'existe aujourd'hui pour factoriser ce grand nombre en temps polynomial. Pour montrer que l'algorithme RSA est vulnérable à tous types d'attaques, nous allons examiner dans cette partie comment un adversaire peut essayer d'obtenir le message clair sans avoir la clé de décryptage.

L'attaque passive

La tâche principale d'un adversaire passif est celui de la récupération de plaintext m à partir de son ciphertext correspondant c, en donnant l'information publique (n, e), cela est exactement le problème RSA donc on peut dire que RSA est sécurisé contre une attaque passive si et seulement si le problème RSA est difficile.

La sécurité sémantique

en fait RSA est un algorithme déterministe et tout les algorithmes déterministes ne sont pas sémantiquement sûrs, on exprime ce point en utilisant ce scénario :

Supposons que l'adversaire sait que l'utilisateur chiffre l'un des deux messages m_1 ou m_2, et il est supposé connaître n et e (la clé publique de l'utilisateur). À la réception du ciphertext c l'adversaire veut déterminer si le plaintext correspondant m est égal à m_1 ou m_2. Tout l'adversaire a besoin de faire est calculé :

$$c' = m_1^e \bmod n. \tag{2.1}$$

alors

- Si $c' = c$ alors $m = m_1$,
- Si $c' \neq c$ alors $m = m_2$.

Donc on peut constater que le cryptosystème RSA n'est pas sémantiquement sûr.

La malléabilité du RSA

RSA est malléable en raison de la propriété homomorphique (voir définition 2.2.1).

Définition 2.2.1 *Un cryptosystème est dit homomorphique si nous pouvons déterminer le chiffrement de $m_1 m_2$, sans savoir m_1 ou m_2.*

Concernant le cryptosystème RSA, un plaintext m est chiffré comme $\mathscr{E}_{pk}(m) = m^e \bmod n$, où (e,n) est la clé publique. Un adversaire peut construire un chiffrement mt pour tout t, comme $\mathscr{E}_{pk}(m)t^e \bmod n = (mt)^e \bmod n = \mathscr{E}_{pk}(mt)$.

On peut utiliser la propriété homomorphique pour montrer que RSA est vulnérable à l'attaque adaptative à texte chiffré choisi comme suit :
Supposons que l'adversaire veut décrypter le message $c = m^e \bmod n$. L'adversaire crée le ciphertext $c' = 2^e c$ et demande à son oracle pour déchiffrer c' pour donner m'. L'adversaire peut alors calculer :

$$\begin{aligned}\frac{m'}{2} &= \frac{c'^d}{2} \\ &= \frac{(2^e c)^d}{2} = \frac{2^{ed} c^d}{2} \\ &= \frac{2m}{2} = m.\end{aligned}$$

Cependant, RSA est vulnérable à toutes sortes d'attaques. En particulier L'attaque à texte clair choisi. Heureusement, l'algorithme RSA a été renforcé, grâce à l'utilisation de fonctions de padding avant le chiffrement. Nous appelons ce type de schéma le

padding RSA.

2.3 Padding RSA

Une idée simple pour attendre une sécurité contre l'attaque à texte clair choisi pour le schéma RSA est de concaténer le message avec une suite aléatoire avant le chiffrement, cette concaténation assure que le message m ne tombe pas dans la gamme de plaintext qui peut être connu et qu'un message donné, une fois concaténé, chiffrera à l'un d'un grand nombre de différents ciphertexts possibles.

Un paradigme général de cette approche est présenté dans la construction suivante : Laisser l'expérience $\text{Exp}^{PRSA}_{\mathscr{A},FCgen}(1^\lambda)$ comme avant et laisser ℓ étre une fonction avec $\ell(\lambda) \leq 2\lambda - 2$ pour tout λ. Définir un schéma de chiffrement à clé publique comme suit :

1. $(pk, sk) \leftarrow \mathscr{G}(1^\lambda)$ **:** Choisir p et q, deux nombres premiers distincts, générer la clé publique (n, e) et la clé secrète (n, d), tel que $n = pq$, e coprime avec $\phi(n) = (p-1)(q-1)$ et $d = e^{-1} \bmod \phi(n)$.

2. $c \leftarrow \mathscr{E}_{pk}(m)$ **:** Donner un message $m \in \{0,1\}^{\ell(\lambda)}$, choisir une chaîne aléatoire $r \leftarrow \{0,1\}^{||n||-\ell(\lambda)-1}$, le ciphertext est $c = [(r||m)^e \bmod n]$.

3. $m \leftarrow \mathscr{D}_{sk}(c)$ **:** Donner un message chiffré $c \in \mathbb{Z}^*_n$, calculer $\widetilde{m} = c^d \bmod n$, le message en clair est $m = [\widetilde{m}]_{\ell(\lambda)}$.

2.3.1 La sécurité du padding RSA

La sécurité du chiffrement padding RSA dépend de la fonction ℓ, si $\ell(\lambda) = 2\lambda - O(log\lambda)$ donc avec une recherche par force brute, en utilisant un algorithme polynomial, sur toutes les valeurs possibles de la chaîne aléatoire r (r doit être la plus aléatoire possible), peut être fait dans $2^{O(log\lambda)}$ alors il n'y a aucun moyen de rendre le schéma résultant CPA sécurisé. Quand $\ell(\lambda) = s\lambda$ pour une constante s il est largement admis que padding RSA est CPA sécurisé, si bien sur le problème RSA est difficile mais il n'y a aucune preuve connue.

En 1988 dans le célèbre travail de Werner Alexi et al. [31] ils ont prouvé qu'avec un algorithme polynomial qui peut déterminer les bits les moins significatifs du message à partir du ciphertext et la clé publique avec une probabilité de réussite égale à $1/2 + \epsilon$ (ϵ une fonction non négligeable) le problème RSA sera plus difficile, ils ont également montré que cette méthode peut être généralisée à les j bits les moins significatifs où j égale à $O(logn)$. En conséquence, chiffrer les messages de longueur $O(logn)$ en utilisant le padding RSA est CPA sécurisé. Cependant, il est plutôt inefficace pour chiffrer les messages de cette manière, car la longueur de la chaîne concaténée serait exponentielle à la longueur du message et dans la pratique, un tel schéma n'est pas utilisé, malgré il est prouvé être sûr.

2.4 Le cryptosystème de Goldwasser-Micali

Aucun système cryptographique déterministe ne peut être CPA sécurisé, puisque l'adversaire $\mathscr{A}$ peut toujours demander à son oracle de re-chiffrer m_0 et m_1. Autrement dit, l'algorithme de chiffrement doit être probabiliste dans la nature et pas seulement déterministe pour résister aux attaques à texte clair choisit.

le cryptosystème de Goldwasser-Micali (GM) [32] est un algorithme asymétrique à clé publique, développé par Shafi Goldwasser et Silvio Micali en 1982, ce dernier est le premier cryptosystème à chiffrement probabiliste qui est prouvablement sûr. La preuve que ce cryptosystème est sémantiquement sûr est basée sur l'hypothèse d'intractabilité du problème de la résiduosité quadratique modulo n [33], lorsque la factorisation de n est inconnu, ceci peut être accompli en utilisant la procédure suivante :

1. $(pk, sk) \leftarrow \mathscr{G}(1^\lambda)$ **:** Génère deux grands nombres premiers p et q tel que $p \neq q$, calculer $n = pq$ puis trouver un non-résidu y pour lequel $\left(\frac{y}{p}\right) = \left(\frac{y}{q}\right) = -1$. La clé publique est (n, y) et la clé secrète est la factorisation (p, q).

2. $c \leftarrow \mathscr{E}_{pk}(m)$ **:** Ecrire m comme une chaîne de bits $\{m_0, m_1 \ldots, m_N\}$. Pour chaque bit m_i générer une valeur aléatoire $z < n$, Le ciphertext est la valeur $c_i = z^2 y^{m_i} \mod n$. .

3. $m \leftarrow \mathscr{D}_{sk}(c)$ **:** Donner un message chiffré $(c_i, \ldots, c_N)$ déterminer si la valeur c_i est un résidu quadratique ; si c'est le cas, $m_i = 0$, autrement $m_i = 1$. Retourner en sortie le message $\{m_0, m_1 \ldots, m_N\}$.

2.4.1 La sécurité du cryptosystème de Goldwasser-Micali

Le cryptosystème de Goldwasser-Micali est le premier schéma de chiffrement qui atteint la sécurité sémantique contre un adversaire passif sous l'hypothèse que la résolution du problème de la résiduosité quadratique est difficile. Pour montrer la sécurité de ce schéma on suppose qu'il existe un adversaire $\mathscr{A}$ contre la sécurité sémantique du chiffrement GM, puis pour déterminer si une valeur x donnée est quadratique résidu modulo n on utilise $\mathscr{A}$ pour voir s'il peut casser le GM en utilisant (y, n) comme une clé publique. Si y est un non-résidu, alors $\mathscr{A}$ doit fonctionner correctement. Cependant, si y est un résidu, donc tous les ciphertexts seront simplement un hasard résidu quadratique. En autre terme la sécurité de GM peut réduire au problème de déterminer si une valeur aléatoire modulo n est un résidu quadratique, ce qui contredirait la difficulté supposée de ce problème si la factorisation de n est inconnue . Cette réduction que l'on a décrit forme la preuve de sécurité sémantiquement du schéma.

Le cryptosystème de GM chiffre un seul bit d'information et la longueur du ciphertext résultant est égale à la longueur du nombre composite n utilisé dans le schéma. Toutefois, il n'est pas efficace car les textes chiffrés peuvent être des centaines des fois plus longues que les textes d'origine.

2.5 Le cryptosystème de Paillier

On reste toujours concentré sur l'attaque à texte clair choisit, un autre cryptosystème probabiliste mérite de le décrire c'est le cryptosystème de Paillier [33]. Cette sec-

tion explore le travail de Paillier en plus ses propriétés intéressantes. Le problème de classe de résiduosité composite est le problème difficile sur lequel se repose ce cryptosystème.

1. $(pk, sk) \leftarrow \mathcal{G}(1^\lambda)$ **:** Choisir deux nombres premiers de grande taille, indépendants et aléatoires : p et q. Sélectionner un entier aléatoire $g \in \mathbb{Z}^*_{n^2}$ où $n = pq$.Assurer que n divise l'ordre de g en vérifiant l'existence de l'inverse multiplicatif modulaire suivante : $\mu = (L(g^\lambda \bmod n^2))^{-1} \bmod n$, où la fonction L est définie comme $L(u) = \frac{u-1}{n}$. La clé publique est (n, g) ; et la clé privée est (λ, μ).

2. $c \leftarrow \mathcal{E}_{pk}(m)$ **:** Obtenir la clé publique (n, g), représenter le message comme un entier m dans l'intervalle $[0, n-1]$. Sélectionner un entier aléatoire $r \in \mathbb{Z}^*_n$, puis calculer le cipertext $c = g^m \cdot r^n \bmod n^2$.

3. $m \leftarrow \mathcal{D}_{sk}(c)$ **:** Pour récupérer m en clair à partir de $c \in \mathbb{Z}^*_n$, calculer le plaintext : $m = L(c^\lambda \bmod n^2) \cdot \mu \bmod n$.

La preuve complète de la validité de déchiffrement de ce système peut être trouvée dans l'article original.

2.5.1 Les propriétés du cryptosystème de Paillier

Le système est un homomorphisme additif (et multiplicatif) ; en d'autres termes, avec uniquement la clé publique et le chiffrement de deux messages m_1 et m_2, il est possible de calculer le chiffrement de $m_1 + m_2$ (et $m_1 m_2$).

L'addition homomorphique

Le produit de deux ciphertexts c'est le ciphertext de la somme de leurs textes en clair correspondants :

$$\mathscr{E}_{pk}(m_1)\mathscr{E}_{pk}(m_2) = (g^{m_1} \cdot r_1^n)(g^{m_2} \cdot r_2^n) \bmod n^2 = g^{m_1+m_2}(r_1 r_2)^n = \mathscr{E}_{pk}(m_1 + m_2) \bmod n^2$$

La multiplication homomorphique

Le chiffrement d'un message à la puissance d'un autre message c'est le ciphertext du produit des deux textes clairs :

$$\mathscr{E}_{pk}(m_1)^{(m_2)} = (g^{m_1} \cdot r^n)^{m_2} \bmod n^2 = g^{m_1 m_2} r^{n m_2} = \mathscr{E}_{pk}(m_1 m_2) \bmod n^2$$

Les propriétés homomorphiques ci-dessus peuvent être utilisées par des nombreux systèmes de sécurité, on peut citer un système important de vote électronique [34] qui assure la confidentialité des votes aussi permet la transparence du scrutin en assurant que l'urne est publique à tout moment ainsi les calculs (comptage..) sont vérifiables à tout, pour plus d'informations on recommande de visiter les références suivantes [35, 36, 37].

2.5.2 La sécurité du cryptosystème de Paillier

Il est conjecturé que le problème de classe de résiduosité composite qui consiste exactement à inverser ce cryptosystème est difficile. La sécurité sémantique est basée

sur la difficulté de distinguer les résidus d'ordre n modulo n^2 des non-résidus d'ordre n. Les résidus d'ordre n modulo n^2 sont les éléments r de $\mathbb{Z}/n^2\mathbb{Z}^*$ tels qu'il existe y tel que $r = y^n \bmod n^2$. Ce problème est en rapport avec l'indistinguabilité des résidus quadratiques dans $\mathbb{Z}/n$.

On note CR[n] le problème de décider les résidus de degré n, i.e. distinguer les résidus de degré n des non-résidus de degré n. La sécurité sémantique du schéma de Paillier avec comme modulo est n est équivalent au problème CR[n]. Le lecteur intéressé par ce problème pourra lire la thèse de Pascal Paillier [38] pour plus de détails.

2.6 Le cryptosystème de Catalano, Gennaro et al

Malgré que le système de Paillier est plus efficace et a des propriétés intéressantes, le coût des opérations pour la multiplication et les exponentiations modulaires en ce qui concerne la génération des clés et le déchiffrement rend ce système plus long. Afin d'accélérer ce dernier Catalano, Gennaro et al., dans [39] ont utilisé un petit élément e tel que e coprime avec $\phi(n)$ au lieu de l'exposant n, comme dans RSA, l'utilisation d'un petit exposant augmente l'efficacité du schéma. Ce dernier aura une meilleure complexité que le système de Paillier, au prix de la perte de l'homomorphie.

1. $(pk, sk) \leftarrow \mathcal{G}(1^\lambda)$ **:** Choisir deux nombres premiers de grande taille, indépendants et aléatoires : p et q. mettre $g = 1 + n$ où $n = pq$. Sélectionner un entier $e \in \mathbb{Z}_n$ tel que $gcd(e, \phi(n)) = 1$. Calcuer l'niverse multiplicatif d de e modulo $\phi(n)$. La clé publique est (n, g, e) ; et la clé privée est (p, q).

2. $c \leftarrow \mathcal{E}_{pk}(m)$ **:** Obtenir la clé publique (n, g, e). Sélectionner un entier aléatoire

$0 < r < n$, puis calculer le cipertext $c = g^m \cdot r^e \bmod n^2$.

3. $m \leftarrow \mathscr{D}_{sk}(c)$ **:** Pour récupérer m en clair à partir de $c \in \mathbb{Z}_n^*$, calcule $r = c^d \bmod n$. On relève ce résultat modulo n^2, puis on récupère g^m en calculant $c/f(r)$ où la fonction f est définie comme $f : x \mapsto (x^e mod n^2)$. On en déduit m par le calcul direct du logarithme discret dans $< g >$.

2.6.1 La sécurité du cryptosystème de Catalano, Gennaro et al

La sécurité sémantique du nouveau schéma peut être prouvée sous une nouvelle hypothèse de classe du petite résiduosité composite qu'est definie comme suit : Étant donné $n = pq$ il est difficile de décider si un élément dans $\mathbb{Z}_{n^2}^*$ est un $e-th$ puissance d'un élément dans $\{0,\ldots,n-1\} \bmod n^2$. Maintenant on peut analyser en profondeur la sécurité de cette nouvelle hypothèse :
Supposons que le système est sémantiquement sécurisé c'est-à-dire pour tous les deux messages m_1 et m_2 $(m_1 \neq m_2)$ il est impossible de distanger d'une maniére efficase entre le chiffrement de m_1 et le chiffrement de m_2. En particulier considérer le cas dans lequel $m_0 = 0$ et $m_1 = 1$. Un chiffrement de m_0 est

$$c_0 = r_0^e \bmod n^2$$

avec $r_0 \in \mathbb{Z}_n$. Observez que c_0 est un petit e-résidu modulo n contrairement au c_1. C' est à cause que $\mathscr{E}_{pk}(m)$ est une permutation . En réalité, puisque chaque chiffrement de zéro est un petit e-résidu, en supposant que c_1 un petit résidu impliquerait qu'il doit être tel que $c_1 = \acute{r}_1^e \bmod n^2$ pour certains $r \in \mathbb{Z}_n$. Dans un tel cas c_1 aurait

deux différentes préimages cela contredit le fait que $\mathscr{E}_{pk}(m)$ est une permutation . Par conséquent, la sécurité sémantique du cryptosystème de de Catalano, Gennaro et al implique que le problème de classe du petite résiduosité composite est difficile.

2.7 Conclusion

La sécurité constitue un grand défi pour construire un schéma robuste à tout type d'attaque, cette problématique constitue les objectifs de cette thèse. Dans ce chapitre, nous avons présenté quelques schémas potentiels et les plus etudiés et cités dans la litterature basés sur le problème de la factorisation en étudiant une notion de sécurité importante dans le cadre des preuves de sécurité de ces schémas est celle de la sécurité sémantique.

Malgré que RSA est souvent utilisé ce dernier est un chiffrement déterministe, et ne peut donc pas être sémantiquement sûr. Une méthode simple pour rendre le chiffrement RSA probabiliste c'est-à-dire sémantiquement sûr, et de concaténer le message avec une suite aléatoire chaque fois avant le chiffrement de manière telle qu'aucune valeur de message, une fois chiffré, ne donne un résultat peu sûr, ceci est appelé le paddind RSA. La sécurité de padding RSA est liée à la suite aléatoire concaténée, qui ne doit jamais être utilisée deux fois pour deux différents messages, cette dernière doit être trop longue pour attendre à ce niveau de sécurité et donc l'inconvénient majeur reste la grande taille des données chiffrées.

Aussi, le premier cryptosystème probabiliste est le système de Goldwasser-Micali basé sur le problème de la résiduosité quadratique. Goldwasser et Micali ont introduit

la notion de sécurité sémantique, qui est toujours utilisée. Ce système est un système de chiffrement inefficace par ce que la taille des textes chiffrés subit une expansion de taille de l'ordre de la taille de sa clef publique, ce qui la rend inutilisable en pratique.

Le cryptosystème de Paillier est un schéma de chiffrement à clé publique sa sécurité est basée sur le problème de classe de résiduosité composite, ce schéma a quelques propriétés d'homomorphe intéressantes utilisées par des nombreuses applications, mais le coût des opérations de ce cryptosystème le rend plus long.

Le cryptosystème de Paillier est réexaminé par Catalano, Gennaro et al, pour démineur sa complexité, la sécurité sémantique de ce nouveau système est maintenant basée sur une nouvelle hypothèse de décision est celle de la dureté de décider si un élément est un "petit" e-ième résidu modulo n^2, ce qui réduit significativement le coût de calcul mais en retour ce nouveau schéma perd les propriétés d'homomorphe.

Cette étude nous a permis de proposer un nouveau cryptosystème probabiliste basé sur la factorisation qui sera présenté dans les prochains chapitres.
La sécurité des cryptosystèmes que l'on va étudier dans le chapitre suivant repose sur un autre problème réputé difficile : le problème du logarithme discret.

Chapitre 3

Cryptosystèmes à clé publique basé sur le logarithme discret

3.1 Introduction

La seconde famille de problèmes cryptographiques est celle des problèmes basés sur le logarithme discret, la raison est que, pour un certain nombre de groupes (les corps finis, les courbes elliptiques...)[40], on ne connaît pas d'algorithmes efficaces pour le calcul du logarithme discret. Avec cette famille Whitfield Diffie et Martin Hellman ont construit l'échange de clé Diffie-Hellman [14]. Le travail de ces derniers qui a servi aux premières constructions cryptographiques à clé publique.

Le schéma de chiffrement ElGamal a été proposé par Taher El Gamal en 1985 [7]et est l'un des quelques systèmes de chiffrement probabiliste basé sur la difficulté de calculer des logarithmes discrets dans un champ fini. La sécurité sémantique du chiffrement ElGamal est en fait équivalente au problème décisionnel de Diffie-Hellman (DDH). Notez que l'obtention de plaintext pour le chiffrement ElGamal est équivalente au problème calculatoire de Diffie-Hellman (CDH). Malheureusement le cryptosystème d'ElGamal n'est pas sûr en raison de sa malléabilité ; en effet, étant donné un chiffré $C = (c_1, c_2)$ pour le message m on peut construire le chiffré $C' = (c_1, tc_2)$, qui sera valide pour le message tm.

Dans ce chapitre nous allons étudier quelques variantes d'ElGamal qui atteignent la sécurité face aux attaques adaptatives à textes chiffrés choisis.

3.2 Le cryptosystème d'ElGamal

La sécurité d'ElGamal est basé sur la difficulté de problème du logarithme discret et le problème de Diffie-Hellman . Ce système est décrit dans cette section.

1. $(pk, sk) \leftarrow \mathcal{G}(1^\lambda)$ **:** Générer un grand nombre premier aléatoire p et un générateur α du group multiplicatif $\mathbb{Z}_p^*$. Sélectionner un entier aléatoire a, $1 \leq a \leq p-2$, et calculer $\alpha^a \bmod p$. La clé publique est (p, α, α^a) ; et la clé privée est a.

2. $c \leftarrow \mathcal{E}_{pk}(m)$ **:** Représenter le message comme un entier m dans l'intervalle $[0, p-1]$. Sélectionner un entier aléatoire k où $1 \leq k \leq p-2$. Calculer $\gamma = \alpha^k \bmod p$ et $\delta = m \cdot (\alpha^a)^k \bmod p$. Le ciphertext est $c = (\gamma, \delta)$.

3. $m \leftarrow \mathcal{D}_{sk}(c)$ **:** Pour récupérer m en clair à partir de c, calculer le plaintext $m = (\gamma)^{-a} \cdot \delta \bmod p$.

3.2.1 La securité d'ElGamal

La fonction à sens unique du cryptosystème ElGamal

Théorème 3.2.1 *Casser le schéma de chiffrement ElGamal, c.-à-d., récupérer m en donnant p, g, g^x, γ et δ, est équivalent à résoudre le problème de CDH .*

preuve. Considérons un adversaire $\mathcal{V}$ qui peut inverser la fonction à sens unique du cryptosystème ElGamal avec une probabilité ϵ. Nous allons montrer que cette probabilité est négligeable.

Nous utilisons d'abord $\mathcal{V}$ pour construire un adversaire $\mathcal{V}^*$ qui calcule la fonction de

Diffie-Hellman :

$\mathcal{V}^*$: Adversaire qui calcule le problème de Diffie-Hellman :

En entrée $(X = g^x, Y = g^y)$, nous devons sortie g^{xy}.

1. Donner X à $\mathcal{V}$ comme une clé publique.

2. Choisir un nombre aléatoire Z et donner (Y, Z) à $\mathcal{V}$ comme un texte chiffré.

3. Lorsque $\mathcal{V}$ retourne m, $\mathcal{V}^*$ retourne Z/m.

Notez que la distribution $(X, (Y, Z))$ en effet correspond à une clé publique de cryptosystème ElGamal et à un chiffrement d'un message aléatoire sous cette clé. Ainsi, avec une probabilité ϵ, $\mathcal{V}$ retourne le correcte plaintext m (si $Z = mg^{xy}$). Lorsque c'est le cas, la sortie du $\mathcal{V}^*$ correspond à la fonction de Diffie-Hellman. Par notre hypothèse que le problème de CDH est difficile dans le group sous-jacent, ϵ doit être négligeable, comme il est désiré. □

La sécurité sémantique du cryptosystème ElGamal

Nous allons montrer que le chiffrement ElGamal a une sécurité sémantique sous l'hypothèse de DDH.

A. **ElGamal est au moins aussi dur que le DDH**

Théorème 3.2.2 *Si le cryptosystème ElGamal n'est pas sémantiquement sécurisé, alors il existe un adversaire qui résout le problème de DDH avec un avantage non négligeable.*

Prouve. Supposons que $\mathscr{A}$ est un adversaire qui peut casser la sécurité sémantique du cryptosystème ElGamal avec un avantage non négligeable ϵ, nous allons l'utiliser pour créer un nouvel adversaire $\mathscr{B}$ qui casse le problème de DDH. La discussion qui suit décrit la construction de $\mathscr{B}$:

Adversaire $\mathscr{B}$:

L'adversaire est donné , p, g, g_1, g_2, g_3 comme entrée (tel que $g_1 = g^x, g_2 = g^y$, et g_3 est soit g^{xy} ou g^z pour certains nombres aléatoire x, y, z).

- Définir $pk = (p, g, g_1)$ et exécuter $\mathscr{A}(pk)$ pour obtenir deux messages m_0, m_1.
- Choisir un bit aléatoire b, et définir $y_1 := g_2$ et $y_2 := g_3 m_b$.
- Donner le ciphertext (y_1, y_2) à $\mathscr{A}$ et obtenir un bit b' comme sortie .

  si $b' = b$ retourner 1 ; sinon retourner 0.

Nous analysons le comportement de $\mathscr{B}$. Il ya deux cas.

Cas 1 ($g_3 = g^{xy}$) : Dans ce cas, (y_1, y_2) est un ciphertext valide de chiffrement ElGamal, alors $\mathscr{A}$ devinera correctement le b avec une probabilité non négligeable. Ainsi :

$$Pr[\mathscr{B} \text{ output=1}] = \frac{1}{2} + \epsilon.$$

Cas 1 ($g_3 = g^z$) : Dans ce cas, b est indépendant du point de vue de l'adversaire, nous affirmons que :

$$Pr[\mathscr{B} \text{ output=0}] = \frac{1}{2}.$$

□

B. **DDH est au moins aussi dur que ElGamal**

Théorème 3.2.3 *Si il existe un d'oracle O qui résout le problème DDH avec une probabilité non négligeable, alors le schéma de chiffrement ElGamal n'est pas sémantiquement sécurisé .*

preuve. Nous supposons que nous avons un oracle O qui résout le problème de DDH, la résolution de ce problème permet l'adversaire $\mathscr{A}$ à distinguer le ciphertext de messages m_0 et m_1.

$\mathscr{A}$ devrait s'exécute en deux étapes :

- **Étape 1 :**Dans cette étape $\mathscr{A}$ demandé à l'oracle de chiffrement ElGamal à chiffrer deux messages m_0, m_1, la sortie de cet oracle est :

$$[g^{k_0}, h^{k_0} m_i], [g^{k_1}, h^{k_1} m_{1-i}], \text{ où } i \in \{0,1\} \text{ et } k_0, k_1 \in \mathbb{Z}_p.$$

- **Étape 2 :** Dans cette étape $\mathscr{A}$ envoyé à l'oracle O la distribution suivante :

$$[g^{k_0}, hg^z, g^{k_0 z} h^{k_0} \frac{m_i}{m_0}], \text{ où } z \in \mathbb{Z}_p.$$

Si la sortie de l'oracle O est $i = 0$ (c.-à-d., un correcte DDH triple) avec une probabilité non négligeable, alors $m_i = m_0$. Sinon, si la sortie de l'oracle O est $i = 1$, également avec une probabilité non négligeable, alors $m_i = m_1$.

Puisque le problème de DDH est difficile dans le group sous-jacent, alors $\epsilon \leq \frac{1}{2}$.

□

La malléabilité d'ElGamal

Cependant même si le cryptosystème ElGamal est sémantiquement sécurisé contre les attaques de texte en clair choisi, ElGamal est trivialement malléable.

prouve. Supposons l'adversaire reçoit le ciphertext :

$$(\gamma, \delta) = (\alpha^k, m \cdot (\alpha^a)^k).$$

Il peut alors créer un ciphertext valide du message $2 \cdot m$ sans jamais savoir m ni le entier k, ni la clé privée x. Le ciphertext qu'il peut produire est donné par :

$$(\gamma, 2 \cdot \delta) = (\alpha^k, 2 \cdot m \cdot (\alpha^a)^k).$$

On peut utiliser cette propriété de malléabilité, pour montrer qu'ElGamal n'est pas sécurisé contre les attaques adaptatives à texte chiffré choisi (CCA2).

prouve. Supposons que l'adversaire veut casser le message suivant :

$$c = (\gamma, \delta) = (\alpha^k, m \cdot (\alpha^a)^k).$$

L'adversaire crée le message associé :

$$c' = (\gamma, 2 \cdot \delta).$$

et demande à son oracle de décryptage pour déchiffrer c "pour donner m". L'adversaire calcule :

$$\frac{m'}{2} = \frac{2\delta\gamma^{-a}}{2} = \frac{2m\cdot(\alpha^a)^k\alpha^{-ak}}{2} = \frac{2m}{2} = m.$$

ElGamal n'est pas robuste

Un objectif principal en cryptographie est la confidentialité des messages, ce qui est généralement obtenu au moyen de chiffrement. En 2010, Abdalla, Bellare et Neven [20] ont soulevé une question essentielle : qu'est-ce qui se passe si un récepteur utilise sa clé secrète sur un ciphertext n'a pas créé sous sa clé publique ? La réalisation de la robustesse n'est pas aussi simple comme elle est parue, ainsi a souligné Abdalla et al [20]. La robustesse est généralement nécessaire aussi bien dans toutes les applications, comme par exemple les protocoles d'enchères avec la confidentialité de l'enchère [41], les systèmes de chiffrement hybrides [42], résultant de la combinaison des composants asymétriques et symétriques ... ect.

Pour bien comprendre l'importance de la propriété de robustesse, la sous section suivante montre comment on peut casser la franchise d'un protocole basé sur le cryptosystème Elgamal, à cause que ce dernier n'est pas robuste.

Une attaque sur la franchise du protocole de Sako

Le protocole de vente aux enchères de Sako [41], il est le premier protocole pratique pour cacher l'enchères des perdants. L'idée de base est la suivante :

1. Soit $V = \{v_1, \ldots, v_n\}$ l'ensemble des valeurs d'enchères possibles. Le commissaire-

priseur prépare n paires de clés (sk_i, pk_i)où $i \in \{1,..,n\}$ et publie les n clés publiques.

2. Pour enchérir pour une valeur v_i un soumissionnaire chiffre un message prédéterminé m sous la clé publique pk_i. Ceci est signé et posté par le soumissionnaire.

3. Pour ouvrir une enchère le commissaire-priseur tente de déchiffrer les enchères cryptées un par un en utilisant sk_n. Si aucun déchiffrage ne retourne m, le commissaire-priseur répète le la procédure en utilisant sk_{n-1}, et ainsi de suite.

Dans [41], Sako a instancié le protocole avec le cryptosystème ElGamal

1. Le commissaire-priseur prépare n paires de clés (a, α^a) où $i \in \{1,..,n\}$, publie le n clés publiques et un message fixé m.

2. Pour enchérir pour une valeur v_i un soumissionnaire choisit $k \in \mathbb{Z}_p$ et poste $(\alpha^a, \alpha^{ak} m)$.

3. Le commissaire-priseur ouvre les enchères selon le protocole et révèle le gagnant.

Si $r = 0$ le chiphertext résultant est de la forme $(1, m)$. Il se déchiffre à m sous n'importe quelle clé secrète. Si un soumissionnaire tricheur et un commissaire-priseur complotent, ils peuvent casser la franchise du protocole.

Ceci est possible car ElGamal n'est pas robuste.

3.3 Le cryptosystème de Damgård

Malgré que le cryptosystème d'ElGamal à été présenté on 1985, seulement en 1998, il a été prouvé que ElGamal est CPA sécurisé [43]. D'autre part, l'ElGamal n'est pas CCA2 sécurisé car il est homomorphique. Cependant, CCA1 sécurité d'ElGamal est un problème ouvert bien connu. On 1991 Damgård a proposé une modification sur le schéma de base d'ElGamal pour attendre CCA1 sécurité mais ce dernier reste toujours vulnérable aux attaques adaptatives à texte chiffré choisi. Cette modification à été élaboré comme suit :

1. $(pk, sk) \leftarrow \mathscr{G}(1^{\lambda})$ **:** Générer un grand nombre premier aléatoire p et un générateur α du group multiplicatif $\mathbb{Z}_p^*$. Sélectionner deux entiers aléatoire a et b et calculer $u = \alpha^a \bmod p$ et $v = \alpha^b \bmod p$. La clé publique est $(p, \alpha, \alpha^a, \alpha^b)$; et la clé privée est (a, b).

2. $c \leftarrow \mathscr{E}_{pk}(m)$ **:** Représenter le message comme un entier m dans l'intervalle $[0, p-1]$ et sélectionner un entier aléatoire r. L'algorithme de chiffrement calcule un ciphertext $c = (x, y, z)$ comme suit :

$$x = \alpha^r, y = u^r, z = mv^r.$$

3. $m \leftarrow \mathscr{D}_{sk}(c)$ **:** Pour déchiffrer un texte chiffré c, l'algorithme de déchiffrement calcule m comme suit : si $y = x^a$ alors

$$m = z/x^b.$$

Sinon, l'algorithme de déchiffrement retourne ∇ pour indiquer un ciphertext invalide.

Nous référons le lecteur au référence [44] pour comprendre en détail la sécurité du système de Damgård.

3.4 Le cryptosystéme de Y. Tsiounis et M. Yung

Le premier schéma qui présente des modifications sur le chiffrement ElGamal pour rendre le système résiste aux attaques CCA2, c'est- à- dire rendre le schéma non-malléabilité, c'est le cryptosystème de Y. Tsiounis et M. Yung [43], aussi ces autours sont ceux qui montrent que la sécurité sémantique d'ElGamal équivalente à DDH. L'idée est basée sur l'utilisation d'un système de preuve à divulgation nulle de connaissance de plaintext sous le modèle de l'oracle aléatoire.

1. $(pk, sk) \leftarrow \mathcal{G}(1^\lambda)$ **:** Générer un grand nombre premier aléatoire p et un générateur α du group multiplicatif $\mathbb{Z}_p^*$. Sélectionner un entier aléatoire a, $1 \leq a \leq p-2$, et calculer $\alpha^a \bmod p$. La clé publique est (p, α, α^a) ; et la clé privée est a.

2. $c \leftarrow \mathcal{E}_{pk}(m)$ **:** L'émetteur envoie une preuve à divulgation nulle de connaissance pour le caractère aléatoire utilisé, par exemple il envoie avec le massage chiffré son nom. Sélectionner des entiers aléatoire u, u' et calculer :

$$A = \alpha, B = \gamma^u m, F = \alpha^{u'}.$$

$$ID = \text{Nom, d'autres informations.}$$

$$C = uH(g, A, B, F, ID) + u'.$$

3. $m \leftarrow \mathscr{D}_{sk}(c)$ **:** Le récepteur obtient le ciphertext $[A, B, F, C, ID]$ et le décrypte comme dans le schéma ElGamal d'origine :

$$m = B / A^a.$$

Le récepteur accepte ce chiffrement seulement si l'équation suivante est satisfaite :

$$\alpha^C = A^{H(g,A,B,F,ID)} F.$$

Sinon, l'algorithme de déchiffrement retourne ∇ pour indiquer un ciphertext invalide.

3.4.1 La sécurité du cryptosystème de Y. Tsiounis et M. Yung

La prouve de sécurité de ce schéma est basé sur le théorème du Bellare et Sahai [45] qui dit qu'un schéma de chiffrement est IND-CCA2 sécurisé si et seulement s' il est non malléable. La preuve procède en deux étapes :

1. D'abord, ils ont montré que la sécurité sémantique est équivalente à DDH, c.-à-d., l'addition de la preuve à divulgation nulle de connaissance n'a aucune influence sur la sécurité sémantique de base du système ElGamal.

2. Puis ils ont supposé que le schéma est malléable pour obtenir une contradiction sur sa sécurité sémantique qu'a été déjà démontrée dans la première étape.

La sécurité a été bien expliquée dans le papier original [43].

La malléabilité est une propriété que peuvent posséder des protocoles cryptographiques. Un cryptosystème est dit malléable s'il est possible de transformer un chiffré d'un message m en un chiffré pour un message $f(m)$ pour une fonction f connue sans connaître le message originel m ni obtenir d'information sur lui.

Le cryptosystème de Y. Tsiounis et M. Yung consiste à attacher au message un code d'authentification de message qui va garantir l'intégrité du chiffré.
Même si le message chiffré en utilisant ce schéma est cinq fois plus grand que le message original ça n'aucune influence négative sur l'efficacité de ce dernier par ce que on peut considérer le système comme un protocole de type « stimulation/réponse » (challenge-response). Le vérificateur et le fournisseur de preuve s'échangent des informations et le vérificateur contrôle si la réponse finale est positive ou négative, au lieu d'envoyer les cinq messages chiffrés à la fois, mais la prouve de sécurité de ce cryptosystème reste toujours sous le modèle de l'oracle aléatoire donc n'a aucune prouve de sécurité dans le modèle standard comme ont décrit les atours dans [46].

3.5 Le cryptosystème de Cramer et Shoup

Cramer et Shoup [22] ont proposé le premier schéma prouvé sûr contre les attaques adaptatives à texte chiffré choisi dans le modèle standard, leur principe est basé sur la difficulté du problème DDH et aussi le schéma nécessite une fonction d'hachage à sens unique [47].

1. $(pk, sk) \leftarrow \mathcal{G}(1^{\lambda})$ **:** Générer un groupe cyclique G d'ordre q avec deux générateurs

aléatoires g_1 et g_2. Choisir cinq valeurs aléatoires $\{x_1, x_2, y_1, y_2, z\}$ de $\{0..q-1\}$, Calculer $c = g_1^{x_1} g_2^{x_2}, d = g_1^{y_1} g_2^{y_2}, h = g_1^z$. La clé publique est (c, d, h) ; et la clé privée est (x_1, x_2, y_1, y_2, z).

2. $c \leftarrow \mathscr{E}_{pk}(m)$ **:** Représenter le message comme un élément m dans G, Sélectionner un entier aléatoire k et calculer :

$$u_1 = g_1^k$$

$$u_2 = g_2^k$$

$$e = h^k m$$

$$\alpha = H(u_1, u_2, e)$$

où H et une fonction d'hachage.

$$v = c^k d^{k^\alpha}.$$

Le ciphertext est (u_1, u_2, e, v).

3. $m \leftarrow \mathscr{D}_{sk}(c)$ **:** Pour déchiffrer un texte chiffré c, l'algorithme de déchiffrement calcule m comme suit :

Calculer $\alpha = H(u_1, u_2, e)$ et vérifier que $v = u_1^{x_1} u_2^{x_2} (u_1^{y_1} u_2^{y_2})^\alpha$.Si ce teste échoue l'algorithme de déchiffrement retourne ∇ pour indiquer un ciphertext invalide. Sinon calculer le plaintext $m = e / u_1^z$.

3.6 La sécurité du cryptosystème de Cramer et Shoup

Cramer et Shoup ont démontré que si le problème DDH est difficile à résoudre dans G et que la fonction d'hachage choisie est résistante à la collision donc le schéma décrit ci-dessus est non malléable. La prouve de ces auteurs est basée sur le jeu décrit dans 1.5.3, Ils ont prouvé qu'aucun adversaire ne peut réussir le jeu avec une probabilité non négligeable. Nous référons le lecteur à [48] pour une preuve détaillée de sa sécurité.

En 2013 dans [49] ont trouvé que le chiffrement ElGamal ainsi toutes ses variantes ne peuvent pas être robustes. La robustesse non seulement aidée à rendre le chiffrement plus résistant à une mauvaise utilisation, mais est également une propriété fondamentalement et importante pour certaines applications de chiffrement à clé publique telle que le cryptage avec des protocoles de recherche [50] ou les protocoles enchères [41]. Par conséquent, il semble être un problème ouvert à proposer une variante simple de chiffrement ElGamal atteindre à la fois la non malléabilité conte les attaques CCA2 et la robustesse sans des hypothèses supplémentaires non-standards.

3.7 Conclusion

Dans ce chapitre, nous avons introduit le concept d'un grand problème, souvent utilisé dans la cryptographie, c'est le problème du logarithme discret. Nous avons présenté aussi quelques systèmes de base basés sur ce problème.

L'objectif principal de cette thèse est d'étudier profondément la sécurité des schémas à clé publique basés sur l'infaisabilité de résoudre le problème de la factorisation

et le problème du logarithme discret.

Toutes ces observations nous ont conduites à apporter des contributions en proposant deux schémas :

1. Le premier schéma introduit un nouveau problème appelé le problème de trouver l'ordre d'un élément dans un group (*Order Finding Problem* (OFP)), nous allons proposer, à partir de ce problème, deux nouveaux systèmes le premier pour le chiffrement et le second pour la signature, la sécurité de cette méthode repose sur le fait que ce problème devient impossible à résoudre calculatoirement pour n un produit de deux nombres premiers suffisamment grands.

2. Le deuxième schéma propose ce qui nous appelons un schéma de chiffrement ElGamal modifié (*Modified ElGamal Encryption Scheme* (MEGES)), et on va prouver qu'il est sécurisé contre une attaque adaptative à texte chiffré choisi dans le modèle standard, aussi nous prouverons que le MEGES atteint l'anonymat ainsi que la propriété de la robustesse forte.

Ces contributions seront détaillées dans les chapitres suivants.

Nos Contributions

Chapitre 4

Un nouveau système de chiffrement à clé publique et de signature probabilistes basés sur le problème de la factorisation

Les systèmes cryptographiques doivent avoir une fonction à sens unique difficile à inverser et en plus une haute sécurité sémantique [51]. De cette manière, il est intéressant d'élaborer un système cryptographique qui est sémantiquement sécurisé et aussi basé sur une fonction à sens unique dont son inverse est comparable à la factorisation de $n = pq$. Une série de protocoles de chiffrement basés sur le problème de la factorisation ont proposé (Voir [52, 6, 53, 54, 55, 56, 57]), et aussi pour les signatures numériques on cite [58, 59, 60].

Le schéma de chiffrement RSA [6] est l'un du chiffrement à clé publique de base basé sur le problème de la factorisation . Il fonctionne comme suit : générer deux grands nombres premiers aléatoire (et distinct) p et q, et calculer $n = pq$ et $\phi(n) = (p-1)(q-1)$. Sélectionner un entier e aléatoire, $1 < e < \phi(n)$, de telle sorte que $\text{PGCD}(e, \phi(n)) = 1$, puis calculer l'entier unique d, $1 < d < \phi(n)$, telle que $ed \equiv 1 \bmod \phi(n)$. La clé publique est (n, e) ; et la clé privée est d. Pour chiffrer un message m calculer $c = m^e \bmod n$. Le texte chiffré est c, et peut être décrypté en calculant $m = c^d \bmod n$.

Malgré la sécurité du RSA est basée sur la factorisation, il n'est pas sécurisé contre les attaques de type texte clair choisi (IND-CPA) [61], car il est déterministe : Supposons que l'adversaire connaît que l'émetteur chiffre juste l'un des deux messages m_1 ou m_2. En obtenant le ciphertext c, l'adversaire doit déterminer si le texte en clair m correspondant est équivalent à m_1 ou m_2. Tout l'adversaire devez faire est de calculer : $c' = m_1^e \bmod n$, alors, si $c' = c$, l'adversaire connaît que $m = m_1$, ou si $c' \neq c$, l'adversaire connaît que $m = m_2$.

Dans notre première contribution, nous introduisons un nouveau problème difficile à résoudre (Order Finding Problem (OFP)), ce qui est basé sur le problème de

la factorisation des entier, et nous analysons ses applications en cryptographie. Nous proposons à partir de ce problème deux systèmes : le premier système pour le chiffrement et le second pour la signature. En outre, les schémas proposés ; par opposition aux systèmes RSA (chiffrement et signature) ; sont de nature probabiliste et non pas simplement déterministe. Notre système de chiffrement est sémantiquement sécurisé (IND-CPA sécurisé) dans le modèle standard.

On dénote par $ord(x)$ l'order de $x \in \mathbb{Z}$ qu'est égale au plus petit élément z talque $x^z = 1$, on dénote aussi par $< x >$ le sous-group généré par x.

4.1 Applications et résultats

4.1.1 Nombre premier sûr

Définition 4.1.1 *Un nombre premier sûr est un nombre premier de la forme 2p + 1, où p est lui-même un nombre premier.*

4.1.2 Le nouveau problème difficile

Soient a et b deux éléments dans Z_n^*, où $ord(a) = \frac{p-1}{2}$ et $ord(b) = \frac{q-1}{2}$, pour deux grands nombres premiers sûrs p et q. Donné $n = pq$, a et b, l'hypothèse suivante définit le problème OFP comme le problème de trouver l'ordre de a et b c-à-dire trouver p et q.

Hypothès 4.1.1 . *Considérez l'expérience suivante associée à l'algorithme $\mathscr{A}$.*

$Exp^{OFP}_{\mathscr{A},FCgen}(1^\lambda)$

$(p,q) \leftarrow FCgen(1^\lambda)$

$n \leftarrow pq$

$a' \leftarrow \mathbb{Z}_n^*$

$b' \leftarrow \mathbb{Z}_n^*$

Si $a' = a$ et $b' = b$ retourner 1 sinon retourner 0.

L'avantage de $\mathscr{A}$ dans l'expérience ci-dessus est définie comme :

$$Adv_{\mathscr{A}}^{OFP}(\lambda) = |Pr[Exp_{\mathscr{A},FCgen}^{OFP}(1^\lambda) = 1]|. \tag{4.1}$$

L'hypothèse de OFP exprime que $Adv_{\mathscr{A}}^{OFP}(\lambda)$ est négligeable pour tout algorithme $\mathscr{A}$ probabiliste et polynomial.

4.1.3 Le protocole de chiffrement

Cette partie décrit le schéma de chiffrement qui se compose de trois algorithmes :

- $(pk, sk) \leftarrow \mathscr{G}(1^k)$: Sélectionner un RSA-module $n = pq$ où p et q sont des nombres premiers sûrs, et sélectionner $\alpha, \beta \in \mathbb{Z}_n^*$, où $\alpha = \frac{p-1}{2}$ et $\beta = \frac{q-1}{2}$, choisir deux entiers δ et γ tel que $\delta\alpha + \gamma\beta = 1$. Maintenant, sélectionner un élément g d'ordre maximal PPCM $((p-1),(q-1))$, cet ordre est égal à $\frac{(p-1)(q-1)}{2}$, et ensuite mettre $a = g^{q-1} \bmod n$ et $b = g^{p-1} \bmod n$, de sorte que nous obtenons $ord(a) = \alpha$ et $ord(b) = \beta$ (voir Fact 1). La clé publique $pk = (n, a, b)$ et la clé secrète $sk = (p, q)$.

- $c \leftarrow \mathscr{E}_{pk}(m)$: Nous souhaitons à chiffrer un message m. Le ciphertext est $c_1 = a^x m \bmod n$ et $c_2 = b^y m \bmod n$, pour deux grandes valeurs aléatoires $x, y \in \mathbb{Z}_n^*$.

- $m \leftarrow \mathscr{D}_{sk}(c)$: Étant donné le ciphertext (c_1, c_2), sortie $m = c_1^{\delta\alpha} c_2^{\gamma\beta} \bmod n$.

Fact 1

Si $a = g^{q-1}$ (la même chose pour $b = g^{q-1}$) alors l'ordre de a est $\frac{p-1}{2}$:

$$\begin{aligned} a^{\frac{p-1}{2}} &= g^{\frac{(p-1)(q-1)}{2}} \\ &= g^{PPCM(p-1)(q-1)} \\ &= 1. \end{aligned}$$

Preuve de la validité du déchiffrement

Au moment de déchiffrement le récepteur calcule :

$$\begin{aligned} c_1^{\delta\alpha} c_2^{\gamma\beta} &= (a^x)^{\delta\alpha} m^{\delta\alpha} (b^y)^{\gamma\beta} m^{\gamma\beta} \\ &= (a^\alpha)^{\delta x} m^{\delta\alpha} (b^\beta)^{\gamma y} m^{\gamma\beta} \\ &= m^{\delta\alpha} m^{\gamma\beta} \\ &= m^{\delta\alpha+\gamma\beta} \\ &= m. \end{aligned}$$

L'analyse de la sécurité

Cette section présente les résultats de la sécurité du système de chiffrement à clé publique proposé dans la premiére parti de ce chapitre.

- **La fonction à sens unique**

Théorème 4.1.1 *Le schéma de chiffrement proposé fournit une fonction à sens*

unique s'il n'existe pas un adversaire qui peut trouver p et q.

Preuve. Il est somple de voir que si le OFP est facile à resoudre dans $\mathbb{Z}_n^*$, un adversaire peut récupérer la clé privée (c-à-d, p et q) à partir de laquelle la détermination de m est évidente. □

Remarque 4.1.1 *Il est fortement recommandé que les valeurs x et y sont inconnues à l'adversaire et aussi $a^x m > n$ et $b^y m > n$.*

- **La sécurité sémantique (IND-CPA sécurité)**

  Définition 4.1.2 *Sélectionner $n = pq$. Définir la formulation :*

$$donné\ \ a, b, f, g, \in \mathbb{Z}_n^*\ \ déterminer\ si\ \ f \in < a > \ \ et\ \ g \in < b > . \tag{4.2}$$

  Nous appelons cela le problème décisionnel de générateur (the decisional generator problem(DGP)) qui est basé sur OFP.

  – **Le chiffrement à clé publique proposé est au moins aussi dur que le DGP**

    Théorème 4.1.2 *Si le chiffrement à clé publique proposé n'est pas sémantiquement sécurisé, alors il ya un adversaire qui résout le DGP avec un avantage non négligeable.*

    Preuve. Supposons que $\mathscr{A}$ est un adversaire qui peut casser le chiffrement à clé publique proposé dans le sens de IND-CPA avec un avantage non négligeable ϵ, nous allons l'utiliser pour créer un nouvel adversaire $\mathscr{B}$ qui casse

le DGP.

La discussion qui suit décrit la construction de $\mathscr{B}$:

Adversaire $\mathscr{B}$:

* L'adversaire est donné $\mathbb{Z}_n^*, a, b, f, g$ en entrée.
* Définir $pk = (n, a, b)$ et exécuter $\mathscr{A}(pk)$ pour obtenir deux messages m_0, m_1.
* Choisir un bits aléatoires $b \in \{0,1\}$, et définir :

  (a) $c_1 = f m_b \bmod n$.

  (b) $c_2 = g m_b \bmod n$.
* Donner le chiphrtext (c_1, c_2) à $\mathscr{A}$ et obtenir un bit de sortie b'.

  Si $b' = b$ retourner 1 ; sinon retourner 0.

Nous analysons le comportement de $\mathscr{B}$. Il y a deux cas :

Cas 1 $f \in < a >$ et $g \in < b >$: Dans ce cas (c_1, c_2) est un ciphertext valide, alors $\mathscr{A}$ devinera le b correctement avec une probabilité non négligeable. Ainsi :

$$Pr[\mathscr{B} \text{ output=1}] = \frac{1}{2} + \epsilon.$$

Cas 2 $f \notin < a >$ et $g \notin < b >$: Dans ce cas f et g sont des nombres aléatoires, alors b est indépendant du point de vue de l'adversaire, nous affirmons que :

$$Pr[\mathscr{B} \text{ output=0}] = \frac{1}{2}.$$

□

– **Le DGP est au moins aussi dur que le chiffrement à clé publique proposé**

Théorème 4.1.3 *Si il existe un d'oracle O qui résout le problème DGP avec une probabilité non négligeable, alors le schéma de chiffrement à clé publique proposée n'est pas sémantiquement sécurisé .*

preuve. Nous supposons que nous avons un oracle O qui résout le problème de DGP tels que la résolution de ce problème permet l'adversaire $\mathscr{A}$ pour distinguer le ciphertext de messages m_0 et m_1.

$\mathscr{A}$ devrait s'exécute en deux étapes :

* **Étape 1** : Dans cette étape $\mathscr{A}$ demandé à l'oracle de chiffrement à chiffrer deux messages m_0, m_1 telque PGCD$(m_0, \phi(n)) = 1$, la sortie de cet oracle est :

$$[fm_i, gm_i], [fm_{1-i}, gm_{1-i}] \text{où } i \in \{0,1\}.$$

* **Étape 2** : Dans cette étape $\mathscr{A}$ envoyé à l'oracle O la distribution suivante :

$$[fm_i m_0^{-1}, gm_i m_0^{-1}].$$

Si la sortie de l'oracle O est $i = 0$ (c-à-d $f \in < a >$ ou $g \in < b >$) avec une probabilité non négligeable, alors $m_i = m_0$. Sinon, si la sortie de l'oracle O est $i = 1$, également avec une probabilité non négligeable-, alors $m_i = m_1$.

Puisque le problème de OFP est difficile dans $\mathbb{Z}_n^*$ il est difficile de trouver p et q, donc la probabilité de déterminer si oui ou non $f \in < a >$ et $g \in < b >$ est négligeable, ce qui signifie que le système de chiffrement proposé est IND-CPA sécurisé et ceci conclut la preuve. □

4.1.4 Le protocole de signature

Soit m un message que l'émetteur souhaite signer. Il exécute le protocole de signature suivant qui se compose de trois algorithmes.

- **KGen** : Sélectionner un RSA-module $n = pq$ où p et q sont des nombres premiers sûrs, et sélectionnez $\alpha, \beta \in \mathbb{Z}_n^*$, où $\alpha = \frac{p-1}{2}$ et $\beta = \frac{q-1}{2}$, choisir quatre entiers δ , γ, r et s tel que $\gamma sp + \delta rq = 1$. Maintenant, sélectionner un élément g d'ordre maximal PPCM $((p-1),(q-1))$, cet ordre est égal à $\frac{(p-1)(q-1)}{2}$, et ensuite mettre $a = g^{q-1} \bmod n$ et $b = g^{p-1} \bmod n$, de sorte que nous obtenons $ord(a) = \alpha$ et $ord(b) = \beta$. La clé de vérification publique $vk = (\delta, \gamma)$ et la clé secrète de signature $sk = (p, q)$.

- **Signe** : Pour signer un message m, choisir au hasard $\omega \in < a >$ et $\psi \in < b >$. Calculer $c_1 = (\omega h(m))^{s\beta} \bmod n, c_2 = (\psi h(m))^{r\alpha} \bmod n$ et $w = (\omega\psi)^{-1} \bmod n$, où $h(.)$ est une fonction de hachage cryptographique [62]. La signature de m est (c_1, c_2, w).

- **Verify** : Donenr une signature (c_1, c_2, w) sur m. Accepter si $h(m) = c_1^{\gamma} c_2^{\delta} w \bmod n$

Preuve de la validité de vérification

Au moment de la vérification, le récepteur calcule :

$$\begin{aligned} c_1^{\gamma} c_2^{\delta} &= \omega^{\gamma s \beta} h(m)^{\gamma s \beta} \psi^{\delta r \alpha} h(m)^{\delta r \alpha} \\ &= \omega^{\gamma s \beta} \psi^{\delta r \alpha} h(m)^{\gamma s \beta + \delta r \alpha} \\ &= \omega^{\gamma s \beta} \psi^{\delta r \alpha} h(m). \end{aligned} \tag{4.3}$$

et

$$\begin{aligned} \omega\psi &= \omega\psi^{\gamma s \beta + \delta r \alpha} \\ &= \omega^{\gamma s \beta + \delta r \alpha} \psi^{\gamma s \beta + \delta r \alpha} \\ &= \omega^{\gamma s \beta} \psi^{\delta r \alpha}. \end{aligned} \tag{4.4}$$

De 4.3 et 4.4, le récepteur trouve que $h(m) = c_1^{\gamma} c_2^{\delta} w \bmod n$, donc la condition de vérification détient.

4.1.5 L'analyse de la sécurité

Théorème 4.1.4 *Si le schéma de signature proposé est sécurisé, alors il n'existe pas un adversaire qui résout le OFP avec une probabilité non négligeable.*

Preuve. Dans cette section, nous pourrions examiner la sécurité de notre système de signature. On remarque que, parce que l'OFP est dificile à resoudre dans $\mathbb{Z}_n^*$, l'adversaire ne peut pas calculer la clé secrète (p,q) à partir de la clé publique, donc il ne peut pas faire mieux que de choisir ω, ψ, s et r au hasard. L'adversaire doit, à ce point, dé-

terminer $c_1 = (\omega h(m))^{s\beta} \bmod n$ et $c_2 = (\psi h(m))^{r\alpha} \bmod n$ la probabilité de réussite est seulement $\frac{2}{\mathbb{Z}_n^*} \times \frac{2}{n}$, qui est négligeable pour n est grand.

4.1.6 Une variante possible du schéma de chiffrement de base

Une variété possible du système de chiffrement principale peut être écrit comme suit :

- $(pk, sk) \leftarrow (1^k)$: Sélectionner $n = 2t+1$ où n est premier et $t = pq$ pour deux grands nombres premiers sûrs p et q, choisir deux entiers δ et γ tels que $\gamma q + \delta p = 1$. Sélectionner un générateur $g \in \mathbb{Z}_n^*$ puis mettre $a = g^{2p}$ et $b = g^{2q}$, donc nous obtenons $ord(a) = q$ et $ord(b) = p$ (voir Fact 2). La clé publique $pk = (n, a, b)$ et la clé secrète $sk = (p, q)$.

- $c \leftarrow E_p k(m)$: Nous souhaitons à chiffrer un message m. Le ciphertext est $c_1 = a^x m \bmod n$ et $c_2 = b^y m \bmod n$, pour deux grandes valeurs aléatoires $x, y \in \mathbb{Z}_n^*$.

- $m \leftarrow D_s k(c)$: Étant donné le ciphertext (c_1, c_2), sortie $m = c_1^{\delta\alpha} c_2^{\gamma\beta} \bmod n$.

Fact 2

Si $a = g^{2p}$ (la même chose pour $b = g^{2q}$), alors l'ordre de a est q :

$$\begin{aligned} a^q &= g^{2pq} \\ &= g^{2t} \\ &= 1. \end{aligned}$$

4.1.7 Conclusions et recherches supplémentaires

Avec l'utilisation de l'OFP, un nouveau schéma de chiffrement et de signature ont été proposés, qui sont sécurisé contre les attaques IND-CPA. L'avantage principal, par rapport à RSA, est que la plupart des attaques connus pour RSA ne sont pas applicables sur les nouveaux régimes. Comme travaux futurs, nous cherchons à améliorer nos principaux systèmes pour assurer la sécurité dans le sens de NM-CCA2 dans le modèle standard.

Chapitre 5

Sécurité et robustesse d'un schéma de chiffrement ElGamal modifié

5.1 Introduction

Le chiffrement ElGamal, inventé par ElGamal [7], est l'un des nombreux schémas de chiffrement probabiliste à clé publique, où sa sécurité repose sur la difficulté de problème du logarithme discret. Cependant, le chiffrement ElGamal est une fonction à sens unique si et seulement l'hypothèse CDH est difficile [63], et aussi est prouvé sécurisé contre les attaques IND-CPA sous la difficulté de l'hypothèse DDH [43], mais est trivialement vulnérable aux attaques adaptatives à texte chiffré choisi (IND-CCA2) [64]. Wu et Stinson [65] montrent qu'il est conjecturé être sécurisé contre les attaques IND-CCA1, mais il n'y a pas eu de preuve officielle. Damgård dans [66], a proposé un système utilise une légère modification sur ElGamal, où il a ajouté une exponentiation supplémentaire pour refuser les ciphertexts incorrectes. Damgård a prouvé que son régime est IND-CCA1 sécurisé mais sans aucune preuve formelle. Seulement en 2006 Gjøsteen [67], a démontré que le système de Damgård est IND-CCA1 sécurisé, mais ce régime encore reste vulnérable aux attaques IND-CCA2. Pour plus d'information sur la sécurité IND-CCA1 du schéma de chiffrement ElGamal voir [44].

La première variante d'ElGamal sécurisée contre les attaques IND-CCA2 dans l'oracle aléatoire, proposé par Tsiounis et Young [43], elle a été fondé sur l'hypothèse DDH et la signature de Schnorr [68]. Par la suite Shoup et Gennaro, ont proposé deux variantes d'ElGamal, nommés TDH1 et TDH2, ces variantes sont IND-CCA2 sécurisé dans l'oracle aléatoire sous la difficulté d'hypothèse de CDH et DDH respectivement. Elles étaient principalement concernées par le contexte de chiffrement de seuil [69].

Le cryptosystème du Cramer et Shoup [22] a été le premier système efficace prouvablement sécurisé contre les attaques IND-CCA2 dans le modèle standard, sa sécurité est basée sur la difficulté de DDH et l'utilisation d'une fonction de hachage résistante à la collision.

Depuis ces variantes, diverses propositions ainsi que des hypothèses du calcul ont été considérés dans la littérature et prouvées pour réaliser la sécurité IND-CCA2. Nous référons le lecteur aux références : [70], [63], [71], [72]et [49].

Ces variantes de chiffrement ElGamal peuvent être prouvées IND-CCA2 sécurisé sous une hypothèse de calcul utilisant une preuve très compliqué, qui est peut être difficile à suivre, à cause de la complexité des techniques utilisées par les auteurs. Néanmoins, il est montré par Seurin et Treger [49], que le système de chiffrement ElGamal et ses variantes ne sont pas généralement réussi à atteindre la propriété de forte robustesse.

Dans ce travail, nous proposons ce que nous appelons un schéma de chiffrement ElGamal modifié (Dorénavant MEGES) et prouvent qu'il est IND-CCA2 sécurisé dans le modèle standard sous l'hypothèses de DDH et d'une fonction de hachage résistante à la collision. On comparaison avec d'autres versions d'ElGamal, avec des propriétés de sécurité identiques, notre schéma vérifie la confidentialité, l'intégrité et l'authentification, et il est également attrayante en ce qu'il a une preuve de sécurité très simple. Notre concept principal est d'ajouter des informations au ciphertext, cette information est utilisée pour vérifier si un ciphertext donné a été correctement produit par le correcte expéditeur ; si le ciphertext est juste retourne le plaintext, autrement rejeter. Autre objectif de schéma proposé est de parvenir à l'anonymat (ANON) ainsi que la

forte robustesse (SROB).

5.2 Le schéma de chiffrement ElGamal modifié (MEGES)

Le chiffrement ElGamal est sécurisé contre toutes les attaques IND-CPA. Nous introduisons un nouveau schéma de chiffrement à clé publique, $\vec{\Omega} = (\vec{\mathscr{G}}, \vec{\mathscr{E}}, \vec{\mathscr{D}})$, qui est dérivée de chiffrement ElGamal de la manière suivante :

MEGES		
$\vec{K} : (pk, sk) \leftarrow \mathscr{G}(1^\lambda)$	$\vec{E} : c \leftarrow \mathscr{E}_{pk}(m)$	$\vec{D} : m \leftarrow \mathscr{D}_{sk}(c)$
$a \leftarrow \mathbb{Z}_p^*$	$k, r, \alpha, z, s \leftarrow \mathbb{Z}_p^*$	$\bar{m} = y_2 / {y_1}^a$
$h = g^a$	$y_1 = g^k$	$m = [\bar{m}]_r$
$pk = (p, g, h)$	$y_2 = (m \Vert r) h^k$	$\beta' = \beta^a$
$sk = (a)$	$\beta = g^\alpha, \beta' = h^\alpha$	$c = H(y_2, \beta, \beta')$
Retourner (pk, sk)	$\psi = y_1^z \beta^s$	Si $\psi = g^{r\omega} y_1^r \beta^c$
	$c = H(y_2, \beta, \beta')$	alors Retourner m
	$\omega = [k(z-r) + \alpha s - \alpha c] r^{-1}$	sinon Retourner ∇
	Retourner$(y_1, y_2, \psi, \beta, \omega)$	

Où H est une fonction de hachage cryptographique, et $[\bar{m}]_r$ désigne la troncature de la chaîne de bits $\bar{m}$ à ses r bits à gauche.

5.2.1 La preuve de sécurité

La non-malléabilité de MEGES

Nous allons montrer que $\vec{\Omega}$ est sécurisé contre toutes les attaques IND-CCA2 dans le modèle standard.

Hypothès 5.2.1 *Le MEGES est non malléable (c-à-d, IND-CCA2 sécurisé) en supposant que :*

1. *L'hypothèse DDH est difficile dans le group sous-jacent.*
2. *La fonction d'hachage est choisie parmi des universelles fonctions cryptographiques à sens unique.*

Preuve. On suppose qu'il existe un adversaire $\mathscr{A}$ qui permet de distinguer avec un avantage non négligeable entre le chiffrement de deux messages de son choix. Nous illustrons que nous pouvons construire un algorithme B qui résout l'hypothèse DDH.

Soit (h, β, β') sont un triple DDH donné à B pour que B envoie la clé publique (p, g, h) à $\mathscr{A}$ qui délivre deux plaintext m_0 et m_1, puis B choisit $b \in \{0,1\}$ au hasard et définit le ciphertext cible égal à $c^{\#} = (y_1, y_2, \beta, \psi, \omega)$, où :

$$k', r', \alpha', z', s' \leftarrow \mathbb{Z}_p^*.$$

$$y_1 = g^{k'}.$$

$$y_2 = (m_b || r') h^{k'}.$$

$$\beta = g^{\alpha'}, \beta' = h^{\alpha'}.$$

$$\psi = y_1^{z'} \beta^{s'}, c = H(y_2, \beta, \beta').$$

$$\omega = [k'(z' - r') + \alpha' s' - \alpha' c] r'^{-1}.$$

Maintenant, l'adversaire doit décider si $c^{\#}$ est le chiffrement de m_0 ou m_1. Si (h, β, β') est un triple DDH correcte (c.-à-d., $(g^a, g^\alpha, g^{a\alpha})$), alors $c^{\#}$ est un chiffrement valide de m_b, et l'adversaire aura un avantage $Adv_{\mathscr{A}} > \frac{1}{2} + \frac{1}{n^k}$ pour deviner correctement le b, mais quand (h, β, β') est un triple aléatoire, alors $c^{\#}$ est valide que avec une probabilité négligeable. Enfin, l'adversaire va afficher son estimation pour b. Si la proposition est juste, le B sortira 1. Sinon il sortira 0.

Dans ce cas (quand (h, β, β') n'est pas un triple DDH) l'adversaire peut demander à l'oracle de décryptage à décrypter le ciphertext $c' = (y'_1, y'_2, \beta', \psi', \omega')$ qui est différent de $c^{\#} = (y_1, y_2, \beta, \psi, \omega)$ (depuis $c' \neq c^{\#}$) pour la raison de connaître des informations utiles si $c^{\#}$ est le ciphertext de m_0 ou m_1. Donc, l'adversaire pourrait trouver $y'_2 \neq y_2$ tel que $H(y_2, \beta, \beta') = H(y'_2, \beta, \beta')$ et $r' = r$, il pourrait alors définir $c^{\#} = (y_1, y'_2, \beta, \psi, \omega)$, qui peut passer le test de l'oracle de déchiffrement. L'oracle de déchiffrement donnerait à l'adversaire, le plaintext m', à partir de lequel il pourrait calculer m_b. Heureusement, l'adversaire a une probabilité négligeable de trouver $y'_2 \neq y_2$ tel que $H(y_2, \beta, \beta') = H(y'_2, \beta, \beta')$ parce que la fonction d'hachage a été supposée être résistante à la collision.

Aussi en raison du choix aléatoire de r la probabilité d'utiliser le bon r est supérieure délimitée par $\frac{1}{2^{t_0}}$, où t_0 est la longueur binaire de r.

Autrement dit, la seule façon que l'adversaire pourrait obtenir la même valeur de hachage $H(y_2, \beta, \beta')$ est de modifier le triple (y_1, ψ, ω), dans ce dernier cas, les jeux dans la figure 5.1 montrent que l'oracle de déchiffrement rejette $c^{\#}$ parce que l'étape de vérification échoue.

□

Game 1 (modify y_1)	Game 2 (modify ψ)	Game 3 (modify ω)
$\bar{m} = y_2/\acute{y}_1^a$	$\bar{m} = y_2/y_1^a$	$\bar{m} = y_2/y_1^a$
$\acute{m} = [\bar{m}]_{\acute{r}}$	$m = [\bar{m}]_r$	$m = [\bar{m}]_r$
$\acute{\beta} = \beta^a$	$\acute{\beta} = \beta^a$	$\acute{\beta} = \beta^a$
$c = H(y_2, \beta, \acute{\beta})$	$c = H(y_2, \beta, \acute{\beta})$	$c = H(y_2, \beta, \acute{\beta})$
$\psi \neq g^{\acute{r}\omega}\acute{y}_1^{\acute{r}}\beta^c$	$\acute{\psi} \neq g^{r\omega}y_1^r\beta^c$	$\psi \neq g^{r\acute{\omega}}y_1^r\beta^c$
return ∇	return ∇	return ∇

Game 4 (modify y_1, ψ)	Game 5 (modify y_1, ω)	Game 6 (modify ψ, ω)
$\bar{m} = y_2/\acute{y}_1^a$	$\bar{m} = y_2/\acute{y}_1^a$	$\bar{m} = y_2/y_1^a$
$\acute{m} = [\bar{m}]_{\acute{r}}$	$\acute{m} = [\bar{m}]_{\acute{r}}$	$m = [\bar{m}]_r$
$\acute{\beta} = \beta^a$	$\acute{\beta} = \beta^a$	$\acute{\beta} = \beta^a$
$c = H(y_2, \beta, \acute{\beta})$	$c = H(y_2, \beta, \acute{\beta})$	$c = H(y_2, \beta, \acute{\beta})$
$\acute{\psi} \neq g^{\acute{r}\omega}\acute{y}_1^{\acute{r}}\beta^c$	$\psi \neq g^{\acute{r}\acute{\omega}}\acute{y}_1^{\acute{r}}\beta^c$	$\acute{\psi} \neq g^{r\acute{\omega}}y_1^r\beta^c$
return ∇	return ∇	return ∇

Game 7 (modify y_1, ψ, ω)
$\bar{m} = y_2/\acute{y}_1^a$
$\acute{m} = [\bar{m}]_{\acute{r}}$
$\acute{\beta} = \beta^a$
$c = H(y_2, \beta, \acute{\beta})$
$\acute{\psi} \neq g^{\acute{r}\acute{\omega}}\acute{y}_1^{\acute{r}}\beta^c$
return ∇

FIGURE 5.1: Jeux utilisés dans la preuve de la non-malléabilité de MEGES.

L'anonymat de MEGES

Le MEGES peut facilement être vu atteindre l'anonymat sous les attaques de CPA. Cependant, nous pouvons aussi prouver son anonymat sous les attaques CCA2 en uti-

lisant la même preuve de la sécurité IND-CCA2.

La forte Robustesse du MEGES

Pour prouver la forte robustesse (comme formalisé par Abdalla et al [20]) du schéma MEGES, nous devons montrer que si l'adversaire produit un ciphertext $c^{\#}$, où $c^{\#}$ est défini comme le chiffrement de $m^{\#}$ sous la clé pk_0, se dernier gagne si le décryptage de $c^{\#}$ sous les clés sk_0, sk_1 correspondantes à pk_0, pk_1 ne sont pas échoués.

Preuve. Supposons que l'adversaire produit le ciphertext $(y_1, y_2, \beta, \psi, \omega)$ et soit la clé publique et la clé secrète correctes sont pk_0, sk_0 respectivement. Si l'étape de vérification dans la figure 5.2 réussit dans les deux cas, alors nécessairement $H(y_2, \beta, \beta') = H(y_2, \beta, \tilde{\beta})$, ça veux dire que l'adversaire réussit à trouver une collision pour H qui est impossible parce que H supposée être résistante à la collision. En outre, pour passer l'étape de vérification la valeur de r doit être égale à la valeur de r'.
La preuve dans la figure 2 5.2, garantit que les tentatives de décryptage échouent avec une forte probabilité si la clé secrète est mal utilisée et ce qui assure la forte robustesse.
□

5.3 L'analyse comparative

Le tableau suivant donne une comparaison de MEGES avec quelques variantes d'ElGamal :

Game 1 $\vec{D} : m \leftarrow (sk_0, c^{\#})$	Game 2 $\vec{D} : \nabla \leftarrow (sk_1, c^{\#})$
$\bar{m} = y_1 / y_2^{sk_0}$	$\bar{m} = y_1 / y_2^{sk_1}$
$m = [\bar{m}]_r$	$\tilde{m} = [\bar{m}]_{r'}$
$\beta' = \beta^{sk_0}$	$\tilde{\beta} = \beta^{sk_1}$
$c = H(y_2, \beta, \beta')$	$\tilde{c} = H(y_2, \beta, \tilde{\beta})$
$\psi = g^{r\omega} y_1^r \beta^c$	$\psi \neq g^{r'\omega} y_1^{r'} \beta^{\tilde{c}}$
return m	return ∇

FIGURE 5.2: Jeux utilisés dans la preuve de la forte robustesse de MEGES.

Schémas	Taille pk/sk	Sécurité	ANON+SROB	Oracle aléatoire
EGES	1/1	CPA	×	Non
Damgård	2/2	CCA1	×	Non
Lite CS	4/4	CCA1	×	Non
CS	5/5	CCA2	×	Non
TDH1	1/1	CCA2	×	Oui
TDH2	1/2	CCA2	×	Oui
Schnorr-Signed ElGamal	1/1	CCA2	×	Oui
DHIES([73])	1/1	CCA2	×	Oui
MEGES	1/1	CCA2	✓	Non

TABLE 5.2: Comparaison des variantes EGES

Discussions :

Comme montre le tableau ci-dessus, MEGES vérifie à la fois la sécurité contre les attaques IND-CCA2, sous l'hypothèse DDH et de courte taille de la clé publique/secrète qui est mieux que les autres schémas. En comparant avec les schémas IND-CCA2 sécurité, l'analyse de notre schéma est dans le modèle standard, ce qui signifie que sa sécurité ne dépend pas de l'oracle aléatoire. Nous également révélé que le système MEGES, encore accomplit l'anonymat ainsi la forte robustesse, où tous les régimes décrits dans le tableau 5.2 ne vérifient pas cette propriété.

5.4 Conclusion

Dans cette partie, nous avons proposé une nouvelle variante d'ElGamal qui est prouvée sémantiquement sécurisée contre les attaques adaptatives à texte chiffré choisi, c'est à dire sécurisée dans le sens de NM-CCA2, sous l'hypotèse de DDH sans l'utilisation des oracles aléatoires. Nous avons affirmé que notre système est sécurisé contre des adversaires actifs, car il peut détecter les participants malveillants, donc il vérifie la confidentialité, l'intégrité et l'authentification. L'étape de vérification retourne échoue si le ciphertext est décrypté avec une clé secrète erronée qui signifie que le schéma modifié est fortement robuste. Une étude comparative, que nous avons effectué, a montré que notre système est plus efficace par rapport à certains schémas bien connus.

Conclusion générale

L'élément de base dans tout système cryptographique à clé publique c'est : un problème du calcul difficile. La sécurité du système de chiffrement est basée sur le fait que la clé privée peut être calculée à partir de la clé publique que par la résolution de ce problème difficile, qui souvent basé sur des fonctions à sens unique, cette dernière est une fonction facile à calculer dans un sens, mais difficile à calculer dans la direction opposée (trouver son inverse).

Les deux grands problèmes, souvent utilisés dans la cryptographie à clé publique, sont le problème de la factorisation des entiers, qui consiste à trouver les facteurs premiers d'un entier composé $n = pq$, l'un des systèmes qui se basent sur ce problème est le système RSA, et le problème du logarithme discret qui consiste à trouver a en donnant que certains β tel que $\beta = \alpha^a$. Un système de base pour le problème du logarithme discret est le système ElGamal.

La résistance aux attaques adaptatives à texte chiffré choisi c.-à-d. la non malléabilité est une notion essentielle pour prouver la sécurité et l'efficacité d'un schéma cryptographique, où un algorithme de chiffrement à clé publique est dit sécurisé si et seulement s'il est sémantiquement sécurisé contre les attaques adaptatives à texte

chiffré choisi (IND- CCA2).

Dans cette thèse nous avons traité plusieurs questions liées à la sécurité des cryptosystèmes à clé publique. Les travaux de recherche, menés de cette thèse, sont résumés dans ce qui suit :

Dans le premier chapitre, l'accent est mis dans un premier temps sur les problèmes mathématiques difficiles à résoudre, après nous avons présenté en détail les types d'attaques, puis la section suivante a eu pour objet d'analyser les notions de sécurité pour le chiffrement à clé publique afin de choisir le niveau de sécurité adéquat, d'autre part une étude des séparations et les implications entre les principales notions de sécurité sont abordées.

Dans le deuxième chapitre, nous avons illustré brièvement quelques cryptosystèmes dont la sécurité est fondée sur des hypothèses de calcul liées au problème de la factorisation en entier. En particulier, nous avons étudié le cryptosystème RSA, le cryptosystème de Goldwasser-Micali et le cryptosystème de Paillier. Nous avons présenté également leur sécurité.

Ensuite, nous nous sommes intéressés dans le troisième chapitre au problème du logarithme discret et nous avons présenté dans un second temps quelques systèmes de base fondés sur ce problème tel que l'échange de clés Diffie-Hellman, ElGamal et ses variantes.

La dernière partie a présenté nos contributions, cette partie est divisée en deux chapitres : le premier chapitre a introduit un nouveau problème du calcul difficile, basé sur le problème de la factorisation, et nous avons analysé ses applications en cryptographie. Nous vous avons proposé, à partir de ce problème, deux schémas : le premier

pour le chiffrement et le second pour la signature. Les schémas proposés ; sont de nature probabiliste et non pas simplement déterministe. Nous avons montré également que notre système de chiffrement est sémantiquement sécurisé (IND-CPA sécurité) dans le modèle standard. Dans le deuxième chapitre, nous avons proposé une nouvelle variante de cryptosystème ElGamal qui est sécurisé contre tout adversaire passif et actif sous l'hypothèse DDH, nous avons prouvé aussi que le schéma proposé est sécurisé contre les attaques adaptatives à texte chiffré choisi dans le modèle standard.

Liste des publications

- Revues internationales

- K. DJEBAILI, l.MELKEMI, "Security and Robustness of a Modified ElGamal Encryption Scheme", International journal of information and communication technology, United Kingdom (Accepted).

- Conférences internationales avec comité de lecture

- K. DJEBAILI, l.MELKEMI,"Cryptanalysis of discrete logarithms based cryptosystem using continued fraction and the Legender's result",IT40D, Tebessa, Algeria.
- K. DJEBAILI, l.MELKEMI," Encryption approach for images based on Householder reflector scheme and extended Hill cipher techniques",ICATS'15, Annaba, Algeria.
- K. DJEBAILI, l.MELKEMI,"A different encryption system based on the factorization problem", ICCC 2015, Algiers, Algeria.
- K. DJEBAILI, l.MELKEMI, "New Practical and Non-malleable Elgamal Encryption for E-voting Protocol", ICACNS 2015, Madrid, Spain.

- K. DJEBAILI, l.MELKEMI "An Algorithm for Image Encryption Based on Matrix Transformation", IWCA'16, Oran-Algerie

- Conférences nationales avec comité de lecture

- K. DJEBAILI, l.MELKEMI, "On the theories of public key cryptosystem", 2015, Batna, Algeria.

- K. DJEBAILI, l.MELKEMI, "Matrix transformation based on Householder reflector for sharing secret image", 2015, Biskra, Algeria

- Presentation poster

- K. DJEBAILI, l.MELKEMI,"The order of group element problem and application in cryptography",ICCC 2015, Algiers, Algeria.

Bibliographie

[1] Jeffrey Hoffstein, Jill Pipher, Joseph H Silverman, and Joseph H Silverman. *An introduction to mathematical cryptography*, volume 1. Springer, 2008.

[2] Jonathan Katz and Yehuda Lindell. *Introduction to modern cryptography*. CRC Press, 2014.

[3] Atul Kahate. *Cryptography and network security*. Tata McGraw-Hill Education, 2013.

[4] Ian Curry. An introduction to cryptography and digital signatures. 2001.

[5] Charles Rackoff and Daniel R Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *Advances in Cryptology—CRYPTO'91*, pages 433–444. Springer, 1992.

[6] Ronald L Rivest, Adi Shamir, and Len Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2) :120–126, 1978.

[7] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *Advances in Cryptology*, pages 10–18. Springer, 1985.

[8] Kevin S McCurley. The discrete logarithm problem. In *Proc. of Symp. in Applied Math*, volume 42, pages 49–74, 1990.

[9] Igor Shparlinski. Computational diffie-hellman problem. In *Encyclopedia of Cryptography and Security*, pages 240–244. Springer, 2011.

[10] Dan Boneh. The decision diffie-hellman problem. In *Algorithmic number theory*, pages 48–63. Springer, 1998.

[11] Deyan Simeonov. Cs259c, final paper : Discrete log, cdh, and ddh.

[12] Tatsuaki Okamoto and David Pointcheval. The gap-problems : A new class of problems for the security of cryptographic schemes. In *Public Key Cryptography : 4th International Workshop on Practice and Theory in Public Key Cryptosystems, PKC 2001, Cheju Island, Korea, February 13-15, 2001. Proceedings*, page 104. Springer, 2003.

[13] Moni Naor and Moti Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 427–437. ACM, 1990.

[14] Whitfield Diffie and Martin E Hellman. New directions in cryptography. *Information Theory, IEEE Transactions on*, 22(6) :644–654, 1976.

[15] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of computer and system sciences*, 28(2) :270–299, 1984.

[16] Danny Dolev, Cynthia Dwork, and Moni Naor. Non-malleable cryptography. In *SIAM Journal on Computing*. Citeseer, 1998.

[17] David Pointcheval. Computational security for cryptography. 2009.

[18] M Bellare, A Desai, D Pointcheval, and P Rogaway. Relations among notions of security for public-key encryption schemes. 2001.

[19] Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. In *Advances in Cryptology—ASIACRYPT 2001*, pages 566–582. Springer, 2001.

[20] Michel Abdalla, Mihir Bellare, and Gregory Neven. Robust encryption. In *Theory of Cryptography*, pages 480–497. Springer, 2010.

[21] Mihir Bellare and Phillip Rogaway. Random oracles are practical : A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM conference on Computer and communications security*, pages 62–73. ACM, 1993.

[22] Ronald Cramer and Victor Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *Advances in Cryptology-CRYPTO'98*, pages 13–25. Springer, 1998.

[23] Mihir Bellare and Phillip Rogaway. Optimal asymmetric encryption. In *Advances in Cryptology-EUROCRYPT'94*, pages 92–111. Springer, 1995.

[24] Mihir Bellare and Phillip Rogaway. Probabilistic signature scheme, April 25 2006. US Patent 7,036,014.

[25] Jean-Sébastien Coron. On the exact security of full domain hash. In *Advances in Cryptology—CRYPTO 2000*, pages 229–235. Springer, 2000.

[26] Amos Fiat and Adi Shamir. How to prove yourself : Practical solutions to identification and signature problems. 1998.

[27] Russell Impagliazzo and Steven Rudichi. Limits on the provable consequences of one-way permutations. In *Advances in Cryptology-CRYPTO'88 : Proceedings*, volume 403, page 8. Springer, 2003.

[28] Ran Canetti, Oded Goldreich, and Shai Halevi. The random oracle methodology, revisited. *arXiv preprint cs/0010019*, 2008.

[29] Shivangi Goyal. A survey on the applications of cryptography. *International Journal of Engineering and Technology*, 2(3) :352–355, 2012.

[30] Alfred J Menezes, Paul C Van Oorschot, and Scott A Vanstone. *Handbook of applied cryptography*. CRC press, 1996.

[31] Werner Alexi, Benny Chor, Oded Goldreich, and Claus P Schnorr. Rsa and rabin functions : Certain parts are as hard as the whole. *SIAM Journal on Computing*, 17(2) :194–209, 1988.

[32] Shafi Goldwasser and Silvio Micali. Probabilistic encryption & how to play mental poker keeping secret all partial information. In *Proceedings of the fourteenth annual ACM symposium on Theory of computing*, pages 365–377. ACM, 1982.

[33] Pascal Paillier and David Pointcheval. Efficient public-key cryptosystems provably secure against active adversaries. In *Advances in Cryptology-ASIACRYPT'99*, pages 165–179. Springer, 1999.

[34] Mary Bellis. The history of voting machines. *online article],(November 1998)[cited January 29 2004], Available at : http ://inventors. about. com/library/weekly/aa111300b. htm*, 2000.

[35] Larry Hardesty. Cryptographic voting debuts, 2009.

[36] Mark Eldridge. Electronic voting systems. 2013.

[37] Lucie Langer. *Privacy and verifiability in electronic voting*. PhD thesis, TU Darmstadt, 2010.

[38] Pascal Paillier. Cryptographiea clé publique basée sur la résiduosité de degré composite. *These de Doctorat, École Nationale Supérieure des Télécommunications, Paris*, 1999.

[39] Dario Catalano, Rosario Gennaro, Nick Howgrave-Graham, and Phong Q Nguyen. Paillier's cryptosystem revisited. In *Proceedings of the 8th ACM conference on Computer and Communications Security*, pages 206–214. ACM, 2001.

[40] Wenbo Mao. *Modern cryptography : theory and practice*. Prentice Hall Professional Technical Reference, 2003.

[41] Kazue Sako. An auction protocol which hides bids of losers. In *Public Key Cryptography*, pages 422–432. Springer, 2000.

[42] Payman Mohassel. A closer look at anonymity and robustness in encryption schemes. In *Advances in Cryptology-ASIACRYPT 2010*, pages 501–518. Springer, 2010.

[43] Yiannis Tsiounis and Moti Yung. On the security of elgamal based encryption. In *Public Key Cryptography*, pages 117–134. Springer, 1998.

[44] Helger Lipmaa. On the cca1-security of elgamal and damgård's elgamal. In *Information Security and Cryptology*, pages 18–35. Springer, 2011.

[45] Mihir Bellare and Amit Sahai. Non-malleable encryption : Equivalence between two notions, and an indistinguishability-based characterization. *IACR Cryptology ePrint Archive*, 2006 :228, 2006.

[46] Prabhanjan Ananth and Raghav Bhaskar. Non observability in the random oracle model. In *International Conference on Provable Security*, pages 86–103. Springer, 2013.

[47] Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 33–43. ACM, 1989.

[48] Neal Koblitz and Alfred J Menezes. Another look at" provable security". *Journal of Cryptology*, 20(1) :3–37, 2007.

[49] Yannick Seurin and Joana Treger. A robust and plaintext-aware variant of signed elgamal encryption. In *Topics in Cryptology–CT-RSA 2013*, pages 68–83. Springer, 2013.

[50] Michel Abdalla, Mihir Bellare, Dario Catalano, Eike Kiltz, Tadayoshi Kohno, Tanja Lange, John Malone-Lee, Gregory Neven, Pascal Paillier, and Haixia Shi. Searchable encryption revisited : Consistency properties, relation to anonymous ibe, and extensions. *Journal of Cryptology*, 21(3) :350–391, 2008.

[51] David Pointcheval. Practical security in public-key cryptography. In *ICICS 2001, Lecture Notes in Computer Science*. Citeseer, 2002.

[52] Michael O Rabin. Digitalized signatures and public-key functions as intractable as factorization. Technical report, DTIC Document, 1979.

[53] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in cryptology—EUROCRYPT'99*, pages 223–238. Springer, 1999.

[54] Josh D Cohen and Michael J Fischer. *A robust and verifiable cryptographically secure election scheme*. Yale University. Department of Computer Science, 1985.

[55] Kaoru Kurosawa, Yutaka Katayama, Wakaha Ogata, and Shigeo Tsujii. General public key residue cryptosystems and mental poker protocols. In *Advances in Cryptology—EUROCRYPT'90*, pages 374–388. Springer, 1990.

[56] David Naccache and Jacques Stern. A new public key cryptosystem based on higher residues. In *Proceedings of the 5th ACM conference on Computer and communications security*, pages 59–66. ACM, 1998.

[57] Vitalii A Roman'kov. New probabilistic public-key encryption based on the rsa cryptosystem. *Groups Complexity Cryptology*, 7(2) :153–156, 2015.

[58] Jonathan Katz. Signature schemes based on the (strong) rsa assumption. In *Digital Signatures*, pages 87–119. Springer, 2010.

[59] Jan Camenisch and Anna Lysyanskaya. A signature scheme with efficient protocols. In *Security in communication networks*, pages 268–289. Springer, 2002.

[60] Marc Fischlin. The cramer-shoup strong-rsa signature scheme revisited. In *Public Key Cryptography—PKC 2003*, pages 116–129. Springer, 2003.

[61] Dan Boneh et al. Twenty years of attacks on the rsa cryptosystem. *Notices of the AMS*, 46(2) :203–213, 1999.

[62] National institute of standards and technology. `http://www.nist.gov/public-safety-security-portal.cfm`.

[63] David Pointcheval. Chosen-ciphertext security for any one-way cryptosystem. In *Public Key Cryptography*, pages 129–146. Springer, 2000.

[64] Eike Kiltz and John Malone-Lee. A general construction of ind-cca2 secure public key encryption. In *Cryptography and Coding : 9th IMA International Conference, Cirencester, UK, December 16-18, 2003, Proceedings*, volume 2898, page 152. Springer, 2003.

[65] Jiang Wu and Douglas R Stinson. On the security of the elgamal encryption scheme and damgard's variant. *IACR Cryptology ePrint Archive*, 2008 :200, 2008.

[66] Ivan Damgård. Towards practical public key systems secure against chosen ciphertext attacks. In *Advances in Cryptology-CRYPTO'91*, pages 445–456. Springer, 1992.

[67] Kristian Gjøsteen. A new security proof for damgård's elgamal. In *Topics in Cryptology–CT-RSA 2006*, pages 150–158. Springer, 2006.

[68] Claus-Peter Schnorr. Efficient signature generation by smart cards. *Journal of cryptology*, 4(3) :161–174, 1991.

[69] Shafi Goldwasser, S Jarecki, and A Lysyanskaya. Cryptography and information security group research project : Threshold cryptology, 2013.

[70] Eiichiro Fujisaki and Tatsuaki Okamoto. How to enhance the security of public-key encryption at minimum cost. In *Public Key Cryptography*, pages 53–68. Springer, 1999.

[71] Eike Kiltz. Chosen-ciphertext secure key-encapsulation based on gap hashed diffie-hellman. In *Public Key Cryptography–PKC 2007*, pages 282–297. Springer, 2007.

[72] David Cash, Eike Kiltz, and Victor Shoup. The twin diffie–hellman problem and applications. *Journal of Cryptology*, 22(4) :470–504, 2009.

[73] Michel Abdalla, Mihir Bellare, and Phillip Rogaway. The oracle diffie-hellman assumptions and an analysis of dhies. In *Topics in cryptology—CT-RSA 2001*, pages 143–158. Springer, 2001.