UNIVERSITÉ DE LIMOGES
ÉCOLE DOCTORALE Science – Technologie – Santé
FACULTÉ des Sciences et Techniques

Laboratoire de Recherche XLIM

Thèse N°11-2006

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE LIMOGES

Discipline : Mathématiques appliquées Spécialité : Cryptographie

présentée et soutenue

par

Sophie BOUTITON

le 3 avril 2006

Généralisation des schémas GQ2 et Rabin-Williams : Equivalence avec la factorisation des grands nombres

Thèse dirigée par Thierry BERGER

Rapporteurs

Jean-Jacques QUISQUATER
Professeur à l'Université catholique de Louvain, à Louvain-la-Neuve
Jacques STERN
Professeur à l'Ecole Nationale Supérieure de Paris

Examinateurs du jury

Thierry BERGER et Jean-Pierre BOREL (Président du jury)
Professeurs à l'Université de Limoges
Jean-Jacques QUISQUATER
Professeur à l'Université catholique de Louvain, à Louvain-la-Neuve
Marc GIRAULT et Louis GUILLOU
Experts-Emérite à France Télécom Division R&D
Philippe GABORIT
Maître de conférences à l'Université de Limoges

Invités du jury

François ARNAULT
Maître de conférences à l'Université de Limoges
François DAUDÉ
Expert à France Télécom Division R&D

Remerciements

Je remercie les rapporteurs qui m'ont fait l'honneur d'accepter ce rôle : Jacques Stern pour ses conseils sur la correction de la première version du manuscrit, Jean-Jacques Quisquater pour son soutien, ainsi que toute son équipe à Louvain-la-Neuve, en particulier Benoît, Ludovic et Julien.

Cette thèse n'aurait pas vu le jour sans le protocole de sécurité GQ2, dont le co-inventeur Louis Guillou m'a proposé ce projet de recherche à France Télécom Division R&D à Rennes.

Proche de mon bureau, je remercie grandement mon *transistor* préféré, François Daudé pour son temps, ses connaissances et son extrême rigueur. Un grand merci à l'ensemble des membres de l'unité TCA (Techniques de Contrôle d'Accès) pour leur accueil et leur bonne humeur au quotidien : André, Chantal, Claudia G., Chantal J., David, Nicole, Noël, Olivier et Pascal. Je suis heureuse de poursuivre un bout de chemin parmi eux. Je remercie également Marc Girault pour sa collaboration à distance, son soutien et d'avoir accepté de faire partie du jury.

Je remercie ma hiérarchie notée mathématiquement «$2\Pi R$», soit formellement Pierre Février et Pierre Quentel, pour la confiance qu'ils m'ont toujours témoignée au cours de ces quatre dernières années.

A l'Université de Limoges, en plus de mon directeur de thèse Thierry Berger et Philippe Gaborit, je remercie tout particulièrement François Arnault pour son soutien déterminant à mi-parcours de la thèse, ainsi que les gardiens de l'ADDMUL : Nicolas, Samuel, Laurent et Guilhem.

Je n'oublie pas Grégoire pour les illustrations d'Alice et Bob, et me souviens des premiers mois bretons parmi toute la bande de la Cité d'Ys : Alban, Eric, JP et Manu.

Je remercie toute ma famille, en particulier mes parents, qui ont toujours répondu présents dans les moments difficiles. Témoins des moments forts de ma courte vie, ils sont toujours dans mon coeur de creusoise. Je n'oublie pas ma petite Toulousaine de Steffy, mes vieux amis Albi et Nico, et le prochain docteur en info, Manu.

Je vous salue tous chaleureusement et vous remercie d'avoir partagé cette partie de ma vie, en attendant de la poursuivre à vos côtés, pour la plupart.

Table des matières

Table des figures

Liste des tableaux

Notations

$\mathbb{Z}$	Ensemble des entiers relatifs
$\mathbb{Q}$	Ensemble des nombres rationnels
$\mathbb{P}$	Ensemble des nombres premiers impairs
p	Entier premier impair
n	Entier composé
f	Nombre de facteurs qui composent un module n
$(a\|p)/(a\|n)$	Symbole de Legendre/ Symbole de Jacobi
$\|x\|$	Longueur en bits d'un entier x
$\mathbb{Z}_p^*$	Groupe multiplicatif $(\mathbb{Z}/p\mathbb{Z})^*$
$\mathbb{Z}_n^*$	Groupe multiplicatif $(\mathbb{Z}/n\mathbb{Z})^*$
coeff	Ratio (en particulier CM/MM)
$x\|\|y$	Concatenation de deux entiers x et y
$pgcd(x, y)$	Plus grand dénominateur commun entre les entiers x et y
$ppcm(x, y)$	Plus petit multiplicateur commun entre les entiers x et y
$Pr()$	Probabilité de
AES	Advanced Encryption Standard
CESTI	Centre d'Evaluation de la Sécurité des Technologies de l'Information
CM	Nombre de carrés modulaires
CPC	Complexité pour le prouveur (*claimant*)
CPV	Complexité pour le vérifieur
CRT	Chinese Remainder Theorem
DCC	Designs, Codes and Cryptography
DCSSI	Direction Centrale de la Sécurité des Systèmes d'Information
DES	Data Encryption Standard
DFA	Differential Fault Analysis
DPA	Differential Power Analysis
DSS	Digital Signature Standard
ECC	Elliptic curves cryptography
EMV	Europay, Mastercard, Visa
FDH	Full Domain Hash
FSE	Fast Software Encryption
GIE	Groupement d'Intérêt Economique
GQ	Guillou Quisquater
HFE	Hidden Field Equations

IBM	International Business Machine
IEC	International Electronical Commission
ISO	International Organization for Standardization
LNCS	Lecture Notes in Computer Sciences
MD (MD4,MD5)	Message Digest
MM	Nombre de multiplications modulaires
MT	Machine de Turing
NDS	Novell Directory Services
NIST	National Institute for Standards and Technology (US)
NIZK	Non-Interactive Zero-Knowledge
NSA	National Security Agency (US)
PKC	Public Key Cryptography
PKI	Public Key Infrastructure
PIN	Personal Identity Number
PSS	Probabilistic Signature Scheme
RACE	Research and Development in Advanced Communications Technologies in Europe
RFID	Radio-Frequency IDentification
RIPEMD	RACE Integrity Primitives Evaluation Message Digest
RSA	Rivest Shamir Adleman
SHA	Secure Hash Algorithm
WCC	Workshop on Codes and Cryptography

Avant-propos

Cette thèse regroupe mes travaux de recherche accomplis sur les protocoles de sécurité GQ2 et Rabin-Williams.

Le point de départ de cette aventure remonte à ma rencontre avec Louis Guillou lors de mon stage de fin d'année en 2002, dont l'objet était l'étude des performances et l'implémentation sur carte à puce du protocole de sécurité GQ2 (Guillou Quisquater 2), une solution d'authentification et de signature basée sur la factorisation des grands nombres.

Dans le premier chapitre, nous rappelons les notions de cryptographie à clé publique. Les enjeux de la protection de l'information et les solutions utilisées pour répondre à ces besoins sont présentés : le chiffrement, l'authentification et la signature électronique. La structure commune des protocoles d'authentification dynamique à 3 passes est décrite, ainsi que leur preuve générale de sécurité basée sur l'impossibilité pratique de résoudre certains problèmes mathématiques difficiles comme le calcul de la factorisation des grands nombres et celui du logarithme discret. Un intérêt particulier est porté aux protocoles de type Fiat-Shamir étudiés depuis près de 20 ans, et connus pour leur simplicité et leur performance.

Ensuite, nous illustrons les principes généraux du chapitre précédent à l'aide de l'exemple du protocole GQ2. Les origines de la mise en place de cette solution de sécurité dans le monde des cartes bancaires sont également rappelées. Deux aspects sont abordés : ses performances, et sa sécurité théorique et pratique.

Le troisième chapitre répond à une nouvelle problématique d'intégration du protocole GQ2 : comment rendre la génération des clés GQ2 compatibles avec l'ensemble des clés RSA déjà existantes ? Cette nouvelle souplesse, unique à ce protocole, a permis de regarder l'évolution de sa complexité pour l'utilisation d'un module composé de plus de deux facteurs ou de plus larges exposants publics.

Le protocole de signature de Rabin-Williams possède de nombreuses propriétés communes au schéma GQ2. Ainsi, les techniques appliquées à la généralisation de ce dernier ont entrainé une nouvelle vision du protocole de Rabin-Williams. Une structure commune de ce type de protocole a été mise en évidence, et a permis la mise en place d'un protocole simple et efficace, sans contrainte sur les modules à considérer, et généralisable à de plus larges exposants.

La factorisation des grands nombres est le plus connu et le plus éprouvé des problèmes mathématiques difficiles. L'un des domaines de recherche explorés est la validation de la clé publique. Ce dernier chapitre propose une extension d'une preuve existante afin de mettre en évidence la proportion des modules considérés.

Chapitre 1

L'authentification basée sur la factorisation du module

La sécurité de l'information dans le monde actuel est une préoccupation de tous. Le développement des moyens de communication dans notre quotidien entraîne un intérêt grandissant chez les personnes malintentionnées. Ainsi, les cryptologues s'affèrent à anticiper les actes frauduleux en établissant des règles de sécurité, pour échanger des informations de manière sûre via le chiffrement à clé secrète ou publique, pour s'assurer de l'identité de son interlocuteur via le procédé d'authentification, ou pour signer électroniquement un document. Les protocoles cryptographiques sont les outils sur lesquels s'appuient les scientifiques pour mettre en oeuvre cette sécurité basée sur une transmission de la confiance.

Ce premier chapitre présente les définitions et propositions que nous utiliserons tout au long du document. Il regroupe certains rappels mathématiques ainsi que les notions utilisées dans les preuves de sécurité des protocoles d'authentification dynamique. Le Forking Lemma sera ici généralisé.

Il est important de rappeler qu'un système de sécurité infaillible n'existe pas. Comme tout dispositif de sécurité, l'étude du contexte d'intégration est fondamentale : aucun système cryptographique ne résisterait à une puissance de calculs infinie des ordinateurs. Mathématiquement, la fiabilité de nombreux schémas repose sur les problèmes difficiles à résoudre. Le plus connu, et le plus éprouvé historiquement, est le problème de la factorisation des grands nombres.

Dans ce chapitre, nous nous intéressons à une catégorie de protocoles cryptographiques prouvés sûrs : les protocoles interactifs à 3 passes à divulgation nulle de connaissance. Leur mode de construction ainsi que leur preuve de sécurité rigoureuse, justifie l'intérêt d'étudier cette famille de schémas ; en particulier la construction simple des schémas de type Fiat-Shamir.

Dans la suite, nous appliquerons cette preuve de sécurité des protocoles sûrs à divulgation nulle de connaissance de la factorisation des grands nombres, au plus récent des protocoles de type Fiat-Shamir : le protocole GQ2.

1.1 Introduction

1.1.1 La cryptographie moderne

La **cryptologie** ou bien encore appelée l'étude du secret, regroupe 2 domaines : la **cryptographie** et la **cryptanalyse**. Alors que les **cryptographes** veulent protéger un secret, les **cryptanalystes** étudient comment attaquer la protection de ce secret.

Les deux propriétés recherchées dans une information qui va être transmise via un **canal de communication** sont ([Gui04]) :

– La **confidentialité** : propriété d'une information, qui pourrait être interceptée par un tiers, autre que le destinataire légitime, à rester secrète.

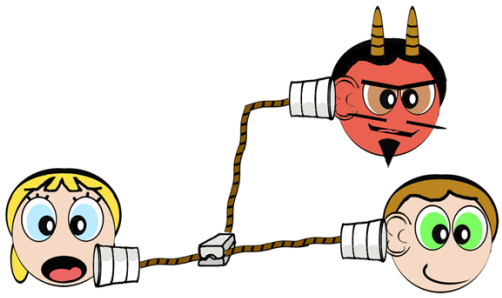


FIG. 1.1 – Problème de confidentialité

– L'**intégrité** : propriété d'une information, qui pourrait être modifiée et réinjectée dans le canal de communication ou dont l'émetteur légitime peut être simulé, à maintenir la provenance et la légitimité de son contenu.

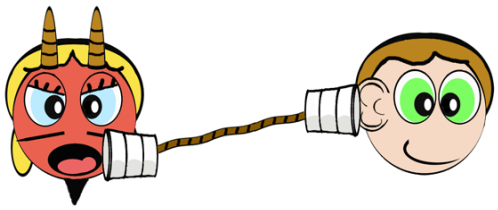


FIG. 1.2 – Problème d'intégrité

Le **message** est le support utilisé pour transmettre l'information (aujourd'hui sous forme numérique quel que soit le contenu : son, image, vidéo...). Ce message est dit **clair** lorsque son contenu n'est pas modifié et peut être accessible à tous.

Le coeur de la cryptologie se matérialise par l'existence de **clés** : séquences de bits spécifiques, qui peuvent être **publiques** ou bien **privées** (**secrètes**). Dans ce dernier cas, une clé permet d'accéder à des informations destinées au détenteur légitime de cette clé : sa longueur est souvent fixée et doit évoluer avec les performances des ordinateurs afin de ne pas être retrouvée au moyen d'une attaque exhaustive (recherche de toutes les clés possibles) par un tiers mal intentionné.

Le premier rôle joué par la **cryptographie à clé secrète** est de répondre au problème de la confidentialité : l'accès aux informations est réservé aux détenteurs d'une clé secrète définie. L'émetteur de l'information chiffre (étape de **chiffrement**), à partir de cette clé, son message clair. Le résultat est appelé **cryptogramme** ou message **chiffré**. Le destinataire rétablit le message clair lors de l'étape de **déchiffrement** à partir de la clé qu'il possède de droit. Si cette restitution est effectuée de manière frauduleuse par une entité n'ayant pas accès légitimement à cette clé secrète, on appelle cette étape le **décryptement** (Remarque : le terme inverse de cette opération n'existe pas).

On schématise ces différents étapes sur la Figure 1.3, où X est le message clair, Y le message chiffré, K la clé, C_K l'opération de chiffrement et D_K l'opération de déchiffrement.

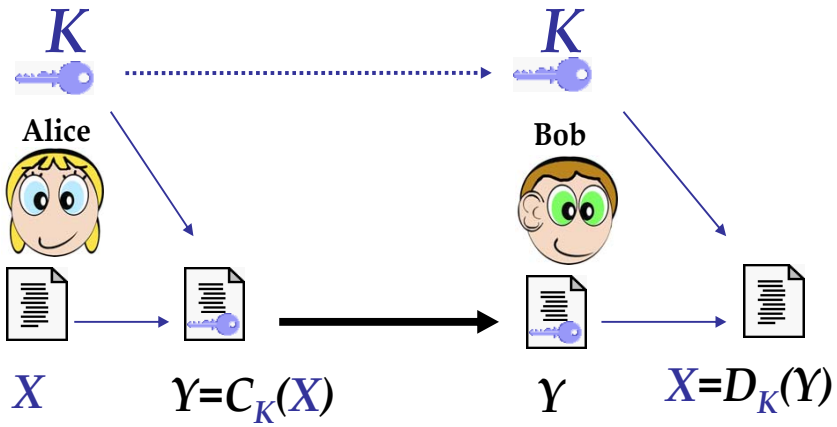


FIG. 1.3 – Chiffrement à clé secrète

La **cryptographie à clé publique** évite le partage d'un secret entre les deux interlocuteurs. Elle repose sur la notion essentielle de l'existence de fonctions à sens unique avec trappe. Le terme **à sens unique** signifie qu'elles sont faciles à calculer mais qu'il est irréalisable par calcul de les inverser. Le terme **avec trappe** signifie que le calcul inverse devient facile lorsque l'on possède une information supplémentaire (la trappe). Chaque utilisateur dispose d'un couple de clés : une clé publique qu'il met en général à disposition de tous dans un annuaire, et une clé privée connue de lui-seul. Le **chiffrement à clé publique** est l'une des utilisations de la cryptographie à clé publique.

On schématise ces différentes étapes sur la Figure 1.4, où M est le message clair, C le message chiffré, $B.KP$ et $B.KS$ les clés publique et secrète de Bob, $C_{B.KP}$ l'opération de chiffrement et $D_{B.KS}$ l'opération de déchiffrement.

Les termes **symétrique ou asymétrique** sont souvent utilisés pour désigner ces deux types de chiffrement. Pour illustrer cette différence, il est intéressant de faire une analogie avec les principes du coffre-fort et de la boîte aux lettres ([dVC01]).

Le chiffrement asymétrique est symbolisé par l'utilisation d'une boite aux lettres : pour un ensemble de n personnes qui désirent communiquer entre elles, la crypto-

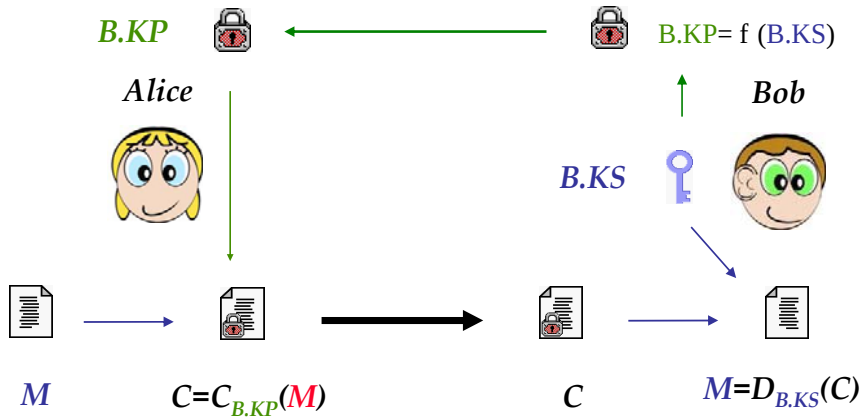


FIG. 1.4 – Chiffrement à clé publique

graphie à clé publique nécessite une génération d'un ensemble de n couples de clés : les clés des n boîtes aux lettres. Par contre, le chiffrement symétrique nécessite un ensemble de $n(n-1)/2$ clés : les clés des $n(n-1)/2$ coffres forts partagées par chacun des couples. Le nombre de clés à produire est exponentiel, et la contrainte de la sécurité lors de la distribution des clés sont les inconvénients de ce mode ; cependant il permet de couvrir à la fois les exigences de confidentialité et d'intégrité des messages transmis.

En plus du chiffrement, l'**authentification** et la **signature électronique** sont également des applications utilisées dans la cryptographie à clé publique. Elles répondent au problème de l'intégrité de l'information.

L'opération d'**authentification** consiste à mettre en relation un **prouveur** (abus de langage) et un **vérifieur**, afin que le premier apporte la preuve au second qu'il est le détenteur d'un secret que lui seul peut connaître, et donc permettre de s'authentifier comme interlocuteur légitime. Dans la suite, on désignera, comme il est de coutume, **Alice** comme le prouveur et **Bob** comme le vérifieur. On est dans le cas d'une **authentification dynamique** lorsqu'un dialogue s'instaure entre les deux entités et que la preuve de la possession d'un secret est calculée en temps réel, et dans le cas d'une **authentification statique**, sinon. Parfois, le terme d'**identification** est utilisé et désigne l'authentification d'une personne.

L'opération de **signature** consiste à mettre en relation un émetteur, appelé **signataire**, et un récepteur, appelé **vérifieur**, que l'on nommera également Alice et Bob. Alice signe un message à partir d'un secret qu'elle seule peut connaître, et Bob vérifie cette signature afin de pouvoir avoir une preuve de l'engagement du signataire dans l'éventualité de désaccords ultérieurs. Juridiquement, cette signature est une preuve équivalente à une signature manuelle. La loi 2000-230 et le décret 2001-272[1], issus de la l'article du code civil 1316 sur la preuve d'engagement d'un tiers, sont une transposition de la directive européenne[2] de 1999, et assure cette légitimité de la preuve électronique.

[1] *http : //archives.internet.gouv.fr*
[2] *http : //www.internet.gouv.fr*

On schématise les différentes étapes de la signature électronique sur la Figure 1.5, où X est le message à signer, $X.S$ la signature du message, $A.KP$ et $A.KS$ les clés publique et secrète d'Alice, $S_{A.KS}$ l'opération de signature et $V_{A.KP}$ l'opération de vérification.

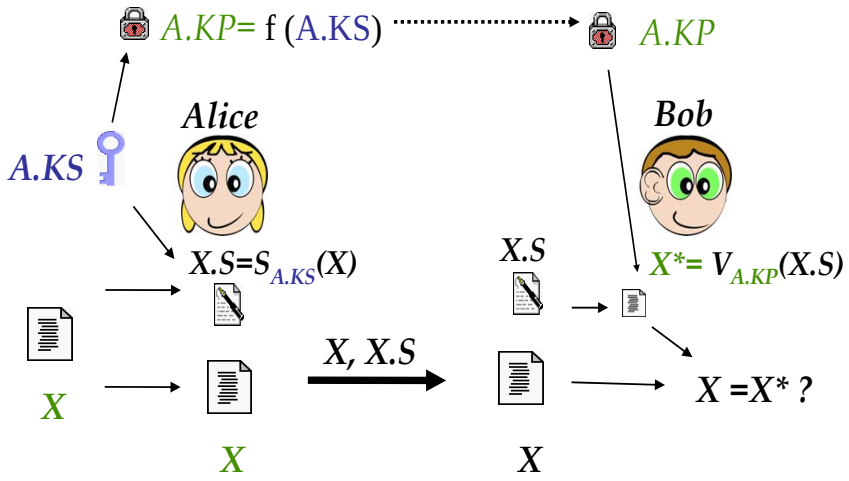


FIG. 1.5 – Signature électronique

Lors de ces étapes de signature, les **fonctions de hachage** sont souvent utilisées afin de parer à certains types d'attaque. Elles permettent d'associer à un message quelconque, une valeur de longueur fixe, appelée **empreinte** ou encore **haché** du message. Cette empreinte peut être assimilée à une valeur pseudo-aléatoire que l'on associe au message. Ces fonctions sont publiques et à sens unique : il n'y a pas de clé secrète et quiconque peut calculer l'empreinte de n'importe quel message, mais, pour une empreinte donnée, il est irréalisable par calcul de trouver un message ayant cette empreinte. Cette propriété à sens unique est issue de la propriété de ces fonctions d'être **à collisions fortes difficiles** ([Sch94]) : il est calculatoirement difficile d'obtenir deux messages différents ayant la même empreinte.

Les **protocoles cryptographiques** sont les outils utilisés pour mettre en oeuvre ces procédés de chiffrement, d'authentification et de signature. Dans le cas de la cryptographie à clé secrète, ce sont des schémas qui reposent sur la protection du message à partir de la clé partagée entre les deux parties (DES, TripleDES, AES, ...) et dans le cas asymétrique, ce sont des schémas qui reposent sur des problèmes mathématiques difficiles.

Le célèbre protocole cryptographique asymétrique, RSA ([RSA78]), découvert en 1977, est utilisé dans le cas du chiffrement, de l'authentification et de la signature.

Un quatrième type de protocole peut être ajouté : l'échange de clé, appelé également **mise à la clé**. L'objectif de cette communication est le partage, la mise en commun, d'une clé secrète K entre Alice et Bob, afin de pouvoir échanger par la suite des informations confidentielles. Le protocole de Diffie-Hellman [DH76] en est un exemple (Figure 1.6), et le problème difficile sur lequel il repose est le suivant :

[DH] : Etant donné un entier p premier, un entier $g \in \mathbb{Z}_p^*$ ainsi que deux éléments $I_1 = g^{s_A} \bmod p$ et $I_2 = g^{s_B} \bmod p$, calculer $g^{s_A \times s_B} \bmod p$.

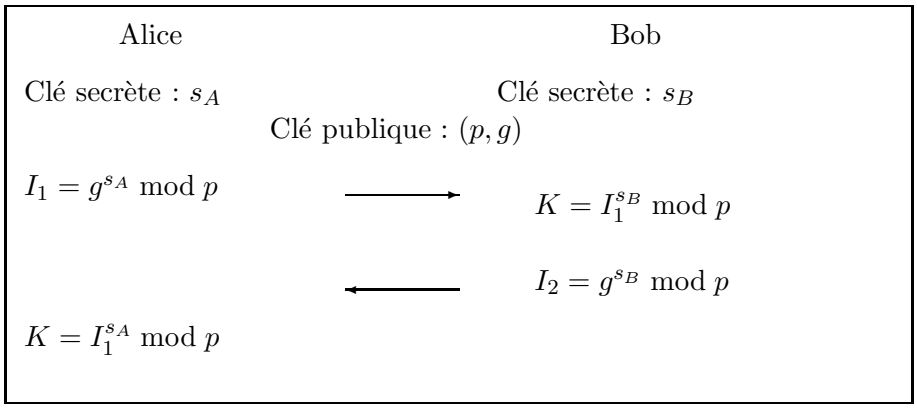


FIG. 1.6 – Protocole de Diffie-Hellman pour une mise à la clé

L'analyse de sécurité d'un système se symbolise par l'étude d'une chaîne d'éléments de **confiance**. Cette chaîne peut se retrouver dans des exemples à grande échelle. En France, il existe un **organisme de certification gouvernemental**, la DCSSI, qui est rattachée au cabinet du Premier Ministre. Les laboratoires d'évaluation, ou CESTI, sont agréés par la DCSSI. Cette chaîne de confiance (gouvernement-DCSSI-CESTI) permet aux entreprises d'utiliser les procédés de sécurité validés par les trois éléments antérieurs de la chaîne. Les utilisateurs de services, par exemple les internautes, sont les derniers maillons de cette chaîne de confiance.

Dans le cas de distribution de clés cryptographiques, l'**autorité de certification** est considérée comme le niveau de sécurité le plus élevé dans cette chaîne de confiance. Dans le cas du chiffrement à clé publique, son rôle est de certifier à tout vérifieur l'authenticité des clés publiques qu'il possède. Par ce biais, il s'assure du lien entre la personne avec laquelle il désire échanger des informations, et la clé publique fournie.

En général, le vérifieur s'adresse à un annuaire pour obtenir le **certificat** contenant la clé publique, et le signataire reçoit la clé privée associée, certifiée également.

1.1.2 Les protocoles cryptographiques à clé publique

De nombreux schémas cryptographiques à clé publique ont été élaborés depuis la présentation des premiers schémas DH et RSA. La naissance de ces schémas est issue de problèmes mathématiques éprouvés difficiles, selon les connaissances actuelles du domaine des mathématiques fondamentales.

La première famille de schémas, la plus ancienne, regroupe les **protocoles arithmétiques** définis par des calculs modulaires : les problèmes de la factorisation des grands nombres et du logarithme discret sont les problèmes difficiles sur lesquels reposent ces schémas.

Le premier problème est celui abordé dans ce document, et le deuxième est lié, par exemple, au protocole d'El Gamal [ElG85] et à une de ses variantes, présentée en 1991 par le NIST : le DSS.

Les **courbes elliptiques** ont fait leur apparition en 1985 ([Mil86] [Kob87]) et permettent de réduire la taille de la clé publique utilisée (160 bits). La trappe du schéma est la connaissance de l'équation d'une courbe, dont les points respectent une loi de groupe : le problème difficile est le calcul du logarithme discret sur cette courbe.

Il existe également la cryptographie basée sur le problème difficile de la recherche d'un plus petit vecteur dans un espace à plusieurs dimensions. Le schéma NTRU proposé dans [HPS98] relie sa sécurité à ce problème.

La **cryptographie multivariable** trouve son inspiration dans le problème difficile dit du «sac à dos ». En particulier, le protocole HFE [Pat96] et le protocole SFlash [CGP01] en sont deux exemples. L'**algorithme LLL** [LLL82] possède des capacités importantes de cryptanalyse pour la résolution de ce problème difficile dans de nombreux cas.

Enfin, la **cryptographie quantique**, ainsi que celle **basée sur les codes correcteurs d'erreurs** sont également des domaines de recherche explorés pour le domaine de la sécurité, bien que leur mise en oeuvre ne soit encore difficile dans les cas réels.

1.2 Notions utiles

Dans cette partie, nous posons les notions mathématiques utilisées dans le document. Les définitions et théorèmes liés à la résiduosité quadratique des éléments d'un groupe multiplicatif, sont rappelés. La mise en évidence du 2-sous groupe de Sylow d'un groupe, permet de mettre en avant l'isomorphisme de groupes relatif à la décomposition unique d'un nombre premier p sous la forme $p = 2^{b_p} p_1 + 1$ où p_1 est impair.

Ces bases théoriques engendrent la mise en place de propositions pour le calcul de l'unique racine d'ordre impair des éléments d'ordre impair dans le groupe considéré, quelle que soit la nature du nombre premier p. Le cas particulier où $p = 3 \bmod 4$ est bien connu, contrairement au cas quelconque qui reste très peu exploité. Ainsi, l'isomorphisme du groupe $\mathbb{Z}_n^*$ où $n = pq$, permet d'étudier le cas général des nombres n composés de deux facteurs premiers. Le théorème de Lenstra est présenté pour généraliser l'exposant public 2, utilisé dans le schéma de GQ2 et de Rabin-Williams.

Nous définissons les machines de Turing et les notations utilisées pour l'estimation de complexité [Pou00], [Poi96] et [Cor01]. Elles sont utilisées par la suite pour établir la preuve de sécurité du chapitre suivant. Quelques attaques physiques sur cartes à puce sont décrites de manière générale.

1.2.1 Résidus quadratiques

Les définitions et théorèmes suivants proviennent de la publication de V. Shoup [Sho04].

Définition 1.2.1 *Pour tout entier positif n, on appelle un entier a un **résidu quadratique** modulo n si $pgcd(a, n) = 1$ et $x^2 = a \bmod n$ pour un entier x. Si tel est le cas, x est appelé une **racine carrée** de a modulo n.*

Théorème 1.2.1 *Pour tout nombre premier impair p, le nombre de résidus quadratiques a modulo p, avec $0 \leq a < p$, est $(p-1)/2$. De plus, si x est une racine carrée de a modulo p, alors $-x$ également, et toute racine carré de a modulo p satisfait $y = \pm x \bmod p$. Alors, pour tout entier $a \neq 0 \bmod p$, on a $a^{(p-1)/2} = \pm 1 \bmod p$, et de plus, a est un résidu quadratique modulo p si et seulement si $a^{(p-1)/2} = 1 \bmod p$.*

Définition 1.2.2 *Pour tout nombre premier impair p et a un entier tel que $pgcd(a,p) = 1$, le* ***symbole de Legendre*** *$(a|p)$ est défini égal à 1 si a est un résidu quadratique modulo p, et -1 sinon. Pour être complet, on définit $(a|p) = 0$ si p divise a.*

Théorème 1.2.2 *(Symbole de Legendre) Soit p un nombre premier impair et soit $a, b \in \mathbb{Z}$. Alors on a :*

1. $(a|p) = a^{(p-1)/2} \bmod p$ *; en particulier,* $(-1|p) = (-1)^{(p-1)/2}$
2. $(a|p)(b|p) = (ab|p)$
3. $a = b \bmod p$ *implique* $(a|p) = (b|p)$
4. $(2|p) = (-1)^{(p^2-1)/8}$
5. *Si q est un nombre premier impair, alors*

$$(p|q) = (-1)^{\frac{p-1}{2}\frac{q-1}{2}}(q|p)$$

Théorème 1.2.3 *(Symbole de Jacobi) Soient m, n deux entiers positifs et soient a, b deux entiers. Alors :*

1. $(ab|n) = (a|n)(b|n)$
2. $(a|mn) = (a|m)(a|n)$
3. $a = b \bmod n$ *implique* $(a|n) = (b|n)$
4. $(-1|n) = (-1)^{(n-1)/2}$
5. $(2|n) = (-1)^{(n^2-1)/8}$
6. $(m|n) = (-1)^{\frac{m-1}{2}\frac{n-1}{2}}(n|m)$

Calcul du symbole de Jacobi [MOV97] : $(a|n)$
Entrée : (a, n) où n entier impair supérieur à 3 et $a \in \mathbb{N}$ tel que $0 \leq a < n$
Si $a = 0$ **alors** retourner (0)
Si $a = 1$ **alors** retourner (1)
Calculer a_0, h tel que $a = 2^h a_0$ avec a_0 impair
Si h est pair **alors** $s \leftarrow 1$ sinon
Si $n = \pm 1 \bmod 8$ alors $s \leftarrow 1$ sinon $s \leftarrow -1$
Si $n = 3 \bmod 4$ et $a_0 = 3 \bmod 4$ **alors** $s \leftarrow -s$
$n_1 \leftarrow n \bmod a_0$
Si $a_0 = 1$ **alors** retourner (s) sinon retourner $(s \times (n_1|a_0))$

Remarque : on note J_n^+ l'ensemble des éléments dont le symbole de Jacobi par rapport à n est $+1$.

Théorème 1.2.4 *(Théorème de Lagrange) Soit n un entier. Si a est un élément d'un groupe fini multiplicatif d'ordre n, alors $a^n = 1 \bmod n$.*

Théorème 1.2.5 *(Petit théorème de Fermat) Pour tout nombre premier p et pour tout entier $a \neq 0 \bmod p$, on a $a^{p-1} = 1 \bmod p$. De plus, pour tout entier a, $a^p = a \bmod p$.*

Définition 1.2.3 *Soit n un entier. L'**indicateur d'Euler** noté $\phi(n)$ est le nombre d'entiers positifs plus petits et relativement premier à n.*

Théorème 1.2.6 *Pour tout nombre premier p et tout entier positif α, $\phi(p^\alpha) = p^{\alpha-1}(p-1)$.*

Théorème 1.2.7 *Si $n = p_1^{\alpha_1}...p_f^{\alpha_f}$ est la factorisation de n en nombres premiers, alors $\phi(n) = \Pi_{i=1}^f p_i^{\alpha_i - 1}(p_i - 1) = n\Pi_{i=1}^f(1 - 1/p_i)$.*

Théorème 1.2.8 *(Théorème d'Euler) Pour tout entier n positif, et tout entier a relativement premier à n, $a^{\phi(n)} = 1 \bmod n$. En particulier, l'ordre multiplicatif de a modulo n divise $\phi(n)$.*

Théorème 1.2.9 *Soit un nombre entier positif n composé de facteurs premiers $n = \Pi_{i=1}^f p_i$. Le nombre de résidus quadratiques a modulo n, avec $0 \leq a < n$ est $\phi(n)/2^f$. De plus, si a est un résidu quadratique modulo n, alors il y a exactement 2^f entiers distincts tels que $x^2 = a \bmod n$ avec $0 \leq x < n$. De plus, un entier a est un résidu quadratique modulo n si et seulement si il est un résidu quadratique modulo p_i pour $i = 1, ..., f$.*

Remarque : on note QR_n l'ensemble des résidus quadratiques modulo n.

1.2.2 Le 2-sous groupe de Sylow

Définition 1.2.4 *Soit G un groupe fini. Il existe un plus petit entier positif m tel que, pour tout $g \in G$, $g^m = 1$. Cet entier est appelé l'**exposant de** G. Il divise l'ordre de G.*

Définition 1.2.5 *Soit G un groupe. L'ensemble des éléments de G dont l'ordre est une puissance de 2 : $\{g \in G \mid \exists m \in \mathbb{N}/g^{2^m} = 1\}$ est un sous groupe de G et est appelé **le 2-sous groupe de Sylow de** G.*

Proposition 1.2.1 *Soit G un groupe cyclique d'ordre n, et soit $m \in \mathbb{N}$. Alors, pour $g \in G$, les deux conditions suivantes sont équivalentes :*
- *Il existe $h \in G$ tel que $g = h^m$*
- *$g^{n/pgcd(m,n)} = 1$*

Décomposition du groupe $\mathbb{Z}_p^*$

Définition 1.2.6 *Soit $p \in \mathbb{P}$. On peut écrire $p - 1$ sous la forme $p - 1 = 2^{b_p}p_1$ avec p_1 impair, de façon unique. On appelle b_p le **niveau de** p et p_1 la **partie impaire de** $p - 1$.*

Remarque : Si $b_p = 1$ alors p est congru à 3 modulo 4.

Proposition 1.2.2 *Soit $p \in \mathbb{P}$ et soit b_p le niveau de p et p_1 la partie impaire de $p-1$. Alors :*

– *Le 2-sous groupe de Sylow de $\mathbb{Z}_p^*$ noté U_p est cyclique d'ordre 2^{b_p}. Ce sous-groupe est l'image de l'application*

$$\begin{array}{rccc} \eta_p : & \mathbb{Z}_p^* & \longrightarrow & \mathbb{Z}_p^* \\ & x & \longmapsto & x^{p_1} \bmod p \end{array}$$

– *L'ensemble des éléments d'ordre impair dans le groupe $\mathbb{Z}_p^*$ noté V_p est un sous-groupe cyclique d'ordre p_1. Ce sous-groupe est l'image de l'application*

$$\begin{array}{rccc} \rho_p : & \mathbb{Z}_p^* & \longrightarrow & \mathbb{Z}_p^* \\ & x & \longmapsto & x^{2^{b_p}} \bmod p \end{array}$$

Preuve. Le groupe $\mathbb{Z}_p^*$ est cyclique d'ordre $2^{b_p}p_1$, alors il se décompose comme le produit d'un groupe cyclique G_2 d'ordre 2^{b_p} et d'un sous groupe cyclique H d'ordre p_1. Or, un élément de G est d'ordre impair si et seulement si il appartient à H. Alors G_2 est le 2-sous groupe de Sylow et H est l'ensemble des éléments d'ordre impair. Donc, on conclut la preuve grâce à la Proposition 1.2.1.

□

Proposition 1.2.3 *En utilisant les notations précédentes, on a l'isomorphisme de groupes suivant :*

$$\begin{array}{rcl} \Phi_p : \mathbb{Z}_p^* & \longrightarrow & U_p \times V_p \\ x & \longmapsto & \big(\eta_p(x), \rho_p(x)\big) \end{array}$$

Preuve. Il est clair que c'est un homomorphisme. Il existe des entiers λ, μ tels que $\lambda p_1 + \mu 2^{b_p} = 1$. Si $\eta_p(x) = \eta_p(y)$ et $\rho_p(x) = \rho_p(y)$ alors, si on éléve la première égalité à la puissance λ et la seconde à la puissance μ, et, que l'on multiplie les résultats, on a $x = y$. Alors Φ_p est injective. Comme les groupes de chaque côté ont le même cardinal, alors Φ_p est même bijective.

□

1.2.3 Racine carrée d'ordre impair des éléments d'ordre impair

Proposition 1.2.4 *Soit $p \in \mathbb{P}$ tel que $p = 3 \bmod 4$ et soit a un entier tel que $0 < a < p$ et $(a|p) = 1$. Alors, x est une racine carrée de a modulo p si et seulement si $x = \pm a^{(p+1)/4} \bmod p$.*

Preuve. Pour montrer que x est un racine carrée de a modulo p, on suppose que $a = \tilde{x}^2 \bmod p$ pour $\tilde{x} \in \mathbb{Z}_p^*$ puisque l'on sait que a est un résidu quadratique modulo p. Alors on a d'après le Théorème 1.2.5 :

$$x^2 = a^{(p+1)/2} = \tilde{x}^{p+1} = \tilde{x}^2 = a \bmod p$$

□

Proposition 1.2.5 *Soit $p \in \mathbb{P}$ et b_p le niveau de p. Soit $a \in \mathbb{Z}_p^*$.*

Si a est d'ordre impair alors $a^{(p-1)/2^{b_p}} = 1 \bmod p$.

Preuve. L'ordre de a divise $p-1$ d'après le Théorème 1.2.5. Par définition de b_p et par l'hypothèse que a est d'ordre impair, on peut également en déduire que l'ordre de a divise $(p-1)/2^{b_p}$, l'ordre impair le plus élevé du groupe.

□

Proposition 1.2.6 *Soit $p \in \mathbb{P}$ tel que $p \neq 3 \bmod 4$ et soit a un entier tel que $0 < a < p$ et $(a|p) = 1$. Si a est d'ordre impair alors x est une racine carrée d'ordre impair de a modulo p si et seulement si $x = a^{(p+2^{b_p}-1)/2^{b_p+1}} \bmod p$ où b_p est le niveau de p.*

Preuve. Par hypothèse, on a $p \neq 3 \bmod 4$ donc p s'écrit sous la forme $p = 2^{b_p}p_1 + 1$ avec p_1 impair et $b_p > 1$. Pour montrer que x est un racine carrée de a modulo p, on suppose que $a = \tilde{x}^2 \bmod p$ pour $\tilde{x} \in \mathbb{Z}_p^*$ puisque l'on sait que a est un résidu quadratique modulo p. Alors on a :

$$x^2 = a^{(p+2^{b_p}-1)/2^{b_p}} = \tilde{x}^{(p+2^{b_p}-1)/2^{b_p}-1} = \tilde{x}^{2((p-1)/2^{b_p}+1)} = a^{(p-1)/2^{b_p}+1} = a \bmod p$$

car $a^{(p-1)/2^{b_p}} = 1 \bmod p$, d'après la Proposition 1.2.5.

□

1.2.4 Isomorphisme du groupe multiplicatif $\mathbb{Z}_n^*$

Soit $n = pq$ avec $p, q \in \mathbb{P}$ et posons

$$p - 1 = 2^{b_p}p_1, \quad q - 1 = 2^{b_q}q_1, \qquad \text{avec } p_1, q_1 \text{ impairs.}$$

On étudie maintenant le groupe $\mathbb{Z}_n^*$ en utilisant la partie précédente et l'isomorphisme chinois Ξ_n :

$$\begin{array}{rccc} \Xi_n : & \mathbb{Z}/n\mathbb{Z} & \longrightarrow & \mathbb{Z}/p\mathbb{Z} \times \mathbb{Z}/q\mathbb{Z} \\ & x & \longmapsto & \big(x \bmod p, x \bmod q\big) \end{array}$$

Proposition 1.2.7 *En utilisant les notations précédentes, on a :*

- *Le 2-sous groupe de Sylow (l'ensemble des éléments d'ordre une puissance de 2) de $\mathbb{Z}_n^*$ noté U_n d'ordre $2^{b_p+b_q}$ et d'exposant 2^b où $b = max(b_p, b_q)$ (cf Définition 1.2.4).*
- *L'ensemble des éléments d'ordre impair dans le groupe $\mathbb{Z}_n^*$ forme un sous-groupe d'ordre p_1q_1. Ce groupe noté V_n est aussi l'ensemble des résidus à la puissance 2^b dans $\mathbb{Z}_n^*$, où $b = max(b_p, b_q)$. Par analogie, on appellera le **niveau de** n, la valeur b.*

Preuve. Selon le théorème des restes chinois et avec les notations définies dans la Proposition 1.2.2, on a la décomposition :

$$\mathbb{Z}_n^* \simeq \mathbb{Z}_p^* \times \mathbb{Z}_q^* \simeq U_p \times V_p \times U_q \times V_q \simeq (U_p \times U_q) \times (V_p \times V_q)$$

Tous les éléments de $U_p \times U_q$ ont pour ordre une puissance de 2 et tous les éléments de $V_p \times V_q$ sont d'ordre impair. Alors $U_p \times U_q$ correspond au 2-sous groupe de Sylow de $\mathbb{Z}_n^*$ et $V_p \times V_q$ correspond à l'ensemble de tous les éléments d'ordre impair, car leur produit forme le groupe entier. D'après la Proposition 1.2.2, ce 2-sous-groupe de Sylow

est le produit d'un groupe cyclique d'ordre 2^{b_p} avec un groupe d'ordre 2^{b_q} ; l'affirmation à propos de l'exposant et de son ordre est donc vraie.

De plus, le groupe $V_p \times V_q$ est d'ordre $p_1 q_1$, le produit des ordres des facteurs. Dans chacun des sous-groupes d'ordre impair V_p et V_q, l'application $x \mapsto x^{2^b}$ est bijective. Mais dans chacun des sous-groupes U_p et U_q, cette application envoie tous les éléments sur l'élément unité. Alors, dans le produit $U_p \times U_q \times V_p \times V_q$, l'image de l'application $x \mapsto x^{2^b}$ est $\{1\} \times \{1\} \times V_p \times V_q$; ce qui correspond au sous-groupe des éléments d'ordre impair comme nous avons vu précédemment.

□

Proposition 1.2.8 *On définit l'application suivante de l'ensemble $\mathbb{Z}_n^*$ dans lui même, en utilisant les notations de la Proposition 1.2.3 :*

$$\eta_n = \Xi_n^{-1} \circ (\eta_p \times \eta_q) \circ \Xi_n, \qquad \rho_n = \Xi_n^{-1} \circ (\rho_p \times \rho_q) \circ \Xi_n$$

Alors, on a :

(a) L'image de η_n est U_n et l'image de ρ_n est V_n.

(b) Ξ_n induit un isomorphisme entre $U_p \times U_q$ et U_n, et entre $V_p \times V_q$ et V_n.

(c) Nous avons un isomorphisme de groupe :

$$\begin{array}{rccl} \Phi_n : & \mathbb{Z}_n^* & \longrightarrow & U_n \times V_n \\ & x & \longmapsto & \big(\eta_n(x), \rho_n(x)\big) \end{array}$$

Preuve. L'affirmation (b) découle de la Proposition 1.2.7. Puis, par construction des applications η_n et ρ_n on en déduit aisément (a). L'existence de Φ_n dans (c) était établie au début de la Proposition 1.2.7.

□

Proposition 1.2.9 *Soit g un élément du groupe $\mathbb{Z}_n^*$. Pour $h \in \mathbb{Z}_n^*$, on a :*

$$h = \eta_n(g) \iff \begin{cases} h = g^{p_1} \bmod p \\ h = g^{q_1} \bmod q \end{cases}$$

Preuve. Par définition de η_n, on a $h = \eta_n(g)$ si et seulement si on a $\Xi_n(h) = \big(\eta_p(g), \eta_q(g)\big)$. D'après l'expression de Ξ_n on voit que c'est le cas si et seulement si les deux équations de droite sont satisfaites.

□

Corollaire 1.2.1 *L'application η_n est un endomorphisme du groupe $\mathbb{Z}_n^*$.*

Lemme 1.2.1 *Si ξ est une racine de l'unité alors $\mathbb{Q}[\xi]$ est un espace euclidien si et seulement si $\mathbb{Z}[\xi]$ est un espace euclidien.*

Théorème 1.2.10 *([Len80]) Si $e = 2, 3, 5, 7, 11$ et ξ une racine e-ième de l'unité telle que $\xi \neq 1 \bmod n$ et $\xi^e = 1 \bmod n$, alors le corps $\mathbb{Q}[\xi]$ est euclidien.*

Remarque : $\mathbb{Z}[i]$ est également euclidien, où i est le nombre complexe qui vérifie $i^2 = -1$ ($e = 4$). Dans leur livre [IR82], K. Ireland et M. Rosen étudient les propriétés de réciprocité quadratique (resp cubique, resp biquadratique), dans les anneaux euclidiens $\mathbb{Z}[\xi]$ où ξ est une racine carrée (resp cubique, resp biquadratique) de l'unité. Les principales notions sont rappelées dans la thèse de M. Joye [Joy97].

1.2.5 Notions associées à la complexité

Définition 1.2.7 *Une **machine de Turing** est une machine abstraite capable d'effectuer n'importe quel calcul. C'est une formalisation d'un « ordinateur ». Une machine de Turing possède un ruban de travail et une fonction de changement d'état qui décrit le « calcul » à effectuer. Pour cela, une tête de lecture/écriture se déplace par impulsions sur ce ruban, composé à l'infini de cases contenant l'un des symboles :* 0 *ou* 1*. Deux symboles supplémentaires* d *et* f *débutent et terminent le programme.*

Définition 1.2.8 *Un **mot** est une séquence de bits du ruban de travail. On note* E^* *l'ensemble de ces mots notés* x *de longueur* $|x|$*.*

Définition 1.2.9 *La **complexité d'une machine de Turing** notée* M*, qui contient initialement le mot* x*, est le nombre d'impulsions nécessaires avant d'atteindre l'état final. On notera* $T_M(x)$ *cette complexité.*

Définition 1.2.10 *Une machine de Turing est dite **polynomiale** (ou **à temps polynomial**) si sa complexité est polynomiale en la taille des données de départ : le nombre d'impulsions avant d'atteindre l'état final est borné.*

$$\exists n_0 \in \mathbb{N}, \exists c \in \mathbb{N} / \forall n \geq n_0, T_M(n) \leq n^c$$

Définition 1.2.11 *Une machine de Turing est dite **probabiliste**, si au cours de son exécution, elle a accès en lecture à un ruban supplémentaire appelé **ruban aléatoire**, contenant une liste de bits aléatoires.*

Notations : L'ensemble des machines de Turing polynomiales probabilistes sera noté MT. On utilise la notation $M_w(x)$ pour désigner le mot calculé par $M \in MT$, à partir du mot initial x et de l'aléa w. Dans la suite du document, toutes les machines de Turing appartiennent à MT.

Définition 1.2.12 *Deux variables aléatoires* $\{U(x)\}$ *et* $\{V(x)\}$ *paramétrées par un ensemble de mots* E^* *(c'est-à-dire à valeurs dans* E^**) sont dites **parfaitement indistinguables** si*

$$\forall x \in E^*, U(x) = V(x)$$

Définition 1.2.13 *Une fonction* f *de* E^* *dans* $\mathbb{R}$ *est dite **négligeable** si :*

$$\forall c \in \mathbb{N}^*, \exists n_0 \in \mathbb{N} / \forall |x| \geq n_0, f(x) < \frac{1}{|x|^c}$$

Remarque : on définit une probabilité **écrasante** P_e, comme la probabilité «complémentaire » d'une probabilité négligeable P_n : $P_e = 1 - P_n$

Définition 1.2.14 *Soit* $P(x, y)$ *un **prédicat à deux variables** calculable en temps polynomial en* $|x|$*. On dit que* y *est **valide** s'il existe* x *tel que* $P(x, y)$*. Pour un* x *fixé, une donnée* y *vérifiant* $P(x, y)$ *est appelée un **témoin** de* x*.*

Notation : Soit S un ensemble fini, on note « $x \leftarrow S()$ » l'algorithme qui assigne x à un élément sélectionné aléatoirement dans S.

Si $P(.,.)$ est un prédicat, la notation $Pr(x \leftarrow S(), y \leftarrow T(), P(x, y))$ désigne la probabilité que le prédicat $P(.,.)$ soit vrai après l'exécution de $x \leftarrow S()$ et $y \leftarrow T()$.

1.2.6 Attaques physiques

L'interactivité des protocoles de sécurité, en particulier ceux intégrés dans les cartes à puce, entraîne la prise en compte de potentielles attaques physiques sur ces composants électroniques. Les attaques DPA et DFA sont les attaques les plus connues et les articles [Koc96][JJK99][BDL97] se rapportent à ces attaques.

Attaque DPA : Differential Power Analysis

Les opérations effectuées par une carte à puce lors de calculs d'exponentiations modulaires sont propices à des attaques physiques sur carte à puce. En particulier, l'algorithme « Square and multiply » et l'algorithme de Montgomery, les plus connus pour les exponentiations modulaires, doivent être implémentés avec précaution, afin de ne pas permettre une attaque par analyse de la variation de la consommation électrique. La protection du secret situé en exposant serait alors compromise.

Square and multiply($C := z^d \bmod n$)
Entrée : z, d, n
$C \leftarrow 1$
Pour i de 0 à $log_2(d)$ **faire**
Si $d_i = 1$ **alors** $C \leftarrow C.z \bmod n$
Faire $C \leftarrow C^2 \bmod n$
fin pour

Les termes de **Timing attack**, **SPA** (Simple Power Analysis), **DPA** (Differential Power Analysis) regroupent ces types d'attaque qui consistent à observer le temps de calcul, la consommation électrique ou la variation de cette consommation, à partir desquels, un attaquant reconstitue le secret (un bit à 1 en exposant induit un temps de calcul plus long qu'un bit à 0).

Attaque DFA : Differential Fault Analysis

La technique des restes chinois permet de réduire les temps de calculs côté prouveur, en effectuant les calculs modulo chacun des facteurs premiers. Cependant, une attaque dite **DFA**, consiste à injecter des fautes en perturbant le déroulement de l'un des deux calculs modulo un des facteurs premiers : la factorisation du module en résulte en appliquant le principe universel (cf Proposition 1.3.1). Cette attaque peut être évitée en vérifiant l'exactitude des données calculées par la carte avant l'envoi au vérifieur. Une contre-attaque consiste à empêcher la carte de faire cette étape de vérification en perturbant son fonctionnement.

Attaque du générateur aléatoire

Bien que les composants actuels soient protégés contre cette attaque, une action frauduleuse consiste à réussir à figer le générateur aléatoire d'une puce électronique. Comme on le voit dans la suite, on déduit la factorisation du module de la connaissance de triplets entrelacés valides.

1.3 Le problème difficile de la factorisation des grands nombres

Dans le cas de la cryptographie asymétrique où la sécurité des protocoles est basée sur le problème difficile de la factorisation du module, la clé publique contient un nombre composé, appelé **module**, produit de grands nombres premiers.

Nous rappelons les problèmes mathématiques équivalents ou liés à la factorisation de ce module, tels que le calcul des racines carrées et le calcul de l'exposant privé RSA. Dans un deuxième temps, nous présentons la situation actuelle des contraintes à respecter pour élaborer ce module, sans faiblesses de sécurité connues.

1.3.1 Problème équivalent à la factorisation des grands nombres

Théorème 1.3.1 *Soit le problème mathématique difficile noté [FACT] défini par « Etant donné un entier n, trouver sa factorisation en un produit de nombres premiers », et soit [SQRT] défini par « Etant donné un entier n et I un élément de $\mathbb{Z}_n^*$, trouver, s'il existe, un élément $S \in \mathbb{Z}_n^*$ tel que $I = S^2 \bmod n$ ».*

Il y a équivalence entre les problèmes [FACT] et [SQRT].

Preuve. Comme nous le verrons par la suite, il est facile de résoudre le problème [SQRT] si le problème [FACT] est résolu (algorithme de Tonelli-Shanks).

Inversement, montrons que si l'on sait résoudre le problème [SQRT], il existe un algorithme capable de calculer la factorisation du module en temps polynomial (cf Définition 1.2.10) :

Factoriser(n)
Entrée : n
Si n est un nombre composé **alors**
  Répéter
    choisir $x \in \mathbb{Z}_n^*$
    calculer $a = x^2 \bmod n$
    calculer une racine carrée y de a modulo n
  jusqu'à ce que $x \neq \pm y \bmod n$
  $n_1 = pgcd(x - y, n)$
  $n_2 = n/n_1$
  Retourner Factoriser n_1, Factoriser n_2

A chaque tour de boucle, il y a 1 chance sur 2 pour que $x \neq \pm y \bmod n$.

Dans le cas où $n = pq$ où p et q sont deux nombres premiers, tout résidu quadratique modulo n possède 4 racines carrées, et chacune d'entre elles diffère de 2 autres, à plus d'un signe près.

□

1.3.2 Problèmes liés à la factorisation des grands nombres

[RSA] (problème RSA) : Etant donné un entier n produit de deux nombres premiers p et q, un entier v premier avec $\phi(n) = (p-1)(q-1)$ et un élément $I \in \mathbb{Z}_n^*$, trouver un $S \in \mathbb{Z}_n^*$ tel que $I = S^v \bmod n$.

Théorème 1.3.2 *Soit (n, v) la clé publique RSA, où $n = pq$. Soit s la clé privée RSA telle que $v.s = 1 \bmod \phi(n)$*

Si on connaît la factorisation de l'entier n, alors on peut calculer s. Réciproquement, si l'on connaît la clé privée RSA s, alors on peut factoriser le module n.

Preuve. En effet, si on connaît la factorisation du module n, on peut calculer $\phi(n) = (p-1)(q-1)$ puis la clé secrète s telle que $s = v^{-1} \bmod \phi(n)$.

Réciproquement, si on connaît s, il existe un algorithme probabiliste permettant d'en déduire la factorisation du module. On pose $k = sv - 1$. Or, k est pair car multiple de $\phi(n)$, et on a également $x^k = 1 \bmod n$ pour tout $x \in \mathbb{Z}_n^*$.

L'élément 1 possède 2 racines carrées non triviales. Ainsi, on a 1 chance sur 2 de révéler la factorisation du module à l'aide du principe universel (cf Proposition 1.3.1) en calculant $pgcd(y-1, n)$, où y parcourt la série $x^{k/2}, x^{k/4}, ...$

□

En 2004, A. May [May04] [CM04] présente le premier algorithme déterministe à temps polynomial qui démontre l'équivalence de la factorisation avec la connaissance de l'exposant privé RSA.

D'autres articles [BDF98] [BM03] [EJMdW05] dans ce domaine s'intéressent à regarder l'existence d'algorithmes à temps polynomial selon une connaissance partielle de l'exposant secret (une partie des bits de poids forts ou une partie des bits de poids faibles), selon des conditions sur la taille de l'exposant public ou privé.

Le problème de l'équivalence du problème RSA avec la factorisation du module, demeure ouvert : le calcul d'une racine v^{ieme} modulo n est-elle équivalente à la factorisation ? L'article D. Boneh et R. Venkatesan [BV98] annonce que si e est petit, la réponse devrait être « non ».

Un autre problème ouvert est l'équivalence du problème de la résiduosité quadratique avec la factorisation.

[QR] (problème de la résiduosité Quadratique) : Etant donné un entier n produit de deux nombres premiers p et q ainsi qu'un entier $I \in \mathbb{Z}_n^*$ tel que $(I|n) = 1$, déterminer s'il existe $S \in \mathbb{Z}_n^*$ tel que $I = S^2 \bmod n$.

La connaissance de la factorisation permet de déterminer si un entier $I \in \mathbb{Z}_n^*$ tel que $(I|n) = 1$, est un carré modulo n, en calculant les symboles de Legendre de I relativement à p et q. La réciproque reste un problème ouvert.

1.3.3 Challenge RSA

RSA est le plus connu et le plus utilisé des protocoles cryptographiques. Il est à l'origine de la cryptographie moderne. Un challenge, mis en place par la société américaine RSA Security, est lancé à toute la communauté scientifique depuis plusieurs années, pour récompenser les équipes des laboratoires du monde entier capables de trouver la factorisation d'entiers de plus en plus grands. La progression exponentielle de la puissance de calculs des ordinateurs ainsi que l'ingéniosité des chercheurs, permettent de repousser les limites de la sécurité des clés pourtant jugées sûres quelques années auparavant.

En 1994, un nombre utilisé dans un article de 1977 est factorisé. Les auteurs annonçaient plusieurs millions d'années pour décomposer ce nombre de 129 chiffres décimaux. En novembre 2001, Daniel Bernstein [Ber01] a publié un projet d'étude permettant d'améliorer considérablement la rapidité de factorisation des clés RSA et affirme qu'une clé RSA de 3072 bits serait cassée dans le même temps qu'une clé RSA de 1024 bits... cependant les résultats de cette étude restent contreversés compte tenu des coûts engendrés ([LSTT02]). En mai 2005, une équipe de chercheurs, F. Bahr, M. Boehm, J. Franke, et T. Kleinjung, réussit à factoriser un nombre de 663 bits, soit un nombre à 200 chiffres décimaux. Les deux facteurs ont été vérifiés par le laboratoire RSA et ce record reste à battre.

La signification du terme de **grand**, associé aux facteurs des modules, est liée directement à l'évolution de la longueur de ces nombres.

1.3.4 Analyse de la robustesse des facteurs

Le choix de la taille des clés conseillé par la DCSSI est de 1536 bits pour une utilisation du module avant 2010, et de 2048 bits pour une utilisation du module avant 2020. Cependant, en plus de la taille des clés, une étude de la sécurité lors de la génération des facteurs doit être effectuée pour parer aux attaques contre des faiblesses connues.

La première étape consiste à vérifier que les facteurs du module sont bien des nombres premiers. Il existe pour cela les différents tests probabilistes de primalité : les tests de Fermat, de Lucas, de Rabin-Miller. En 2002, un algorithme à temps polynomial nommé AKS du nom de ses inventeurs, est présenté [AKS02].

La deuxième étape tend à vérifier la robustesse de ces facteurs face à différentes méthodes de factorisation du module. Les plus connues sont répertoriées dans [Sti95] et [Arn02] : le critère d'Eratostène, les méthodes de Fermat, de ρ Pollard, les méthodes $p-1$ de Pollard et $p+1$ de Williams, la méthode de Dixon couplée aux méthodes de crible...

Parmi ces méthodes, le **principe universel** est largement utilisé dans ce document.

Proposition 1.3.1 *(Principe universel) Soit n un entier composé de deux facteurs premiers tels que $n = pq$.*

Si deux entiers x et y vérifient $x^2 = y^2 \bmod n$ et $x \neq \pm y \bmod n$ alors $pgcd(x \pm y, n) = p$ ou q.

Preuve. Si $x^2 = y^2 \bmod n$ alors $p|(x-y)(x+y)$. Supposons sans restriction de généralité, que $p|(x-y)$. On a également $p|n$ alors $p|pgcd(n, x-y)$. Or $q \nmid (x-y)$, sinon on aurait par contradiction avec l'hypothèse $x = y \bmod n$. Donc $q \nmid pgcd(n, x-y)$, et ainsi $p = pgcd(n, x-y)$ d'après la décomposition unique en facteurs premiers de n.

□

Un **module sûr** doit être composé de strong facteurs premiers de taille équivalente [LS98] et éloignés les uns des autres ([MOV97]).

Définition 1.3.1 *Un nombre premier p est dit **strong** si des entiers r, s et t existent tels que les conditions suivantes sont satisfaites :*

– $p-1$ *a un grand facteur premier* r
– $p+1$ *a un grand facteur premier* s
– $r-1$ *a un grand facteur premier* t

Remarque : les modules générés dans la pratique, sont sûrs avec une probabilité écrasante.

1.4 Protocoles interactifs à 3 passes à divulgation nulle de connaissance

Les protocoles interactifs à 3 passes ont été définis par [GMR89] afin d'augmenter les performances et accroître la sécurité des schémas d'authentification, grâce au concept de protocoles de sécurité à divulgation nulle de connaissance. Après une présentation de la structure de ces protocoles et une approche intuitive de la propriété de zero-knowledge, nous posons le formalisme utilisé et introduit par M. Bellare [BR93] pour les protocoles zero-knowledge prouvés sûrs. Ces protocoles possèdent une structure identique, et nous nous intéressons en particulier à ceux de type Fiat-Shamir.

1.4.1 Protocoles interactifs à 3 passes : notions

Les protocoles interactifs à 3 passes regroupent les protocoles d'authentification dynamique au cours desquels 3 échanges de données s'opèrent entre le prouveur A et le vérifieur B : l'**engagement** noté W, le **défi** noté d, et la **réponse** notée D.

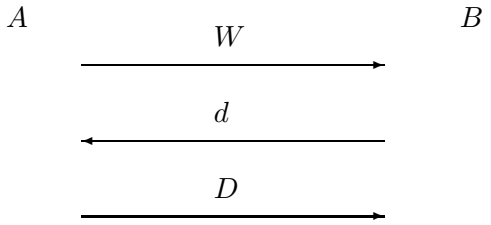


FIG. 1.7 – Protocole à 3 passes

On appelle **triplet** l'ensemble $\{W, d, D\}$. Un triplet est dit **valide** lorsque le prouveur est accepté par le vérifieur. Schématiquement, on a la Figure 1.7. Deux triplets valides $\{W, d, D\}$ et $\{W, d^*, D^*\}$ tels que $d \neq d^*$ seront appelés **triplets entrelacés**.

Dans la suite, on appelle **triplets entrelacés valides** des triplets entrelacés qui permettent de dévoiler la factorisation du module.

L'**équation générique** d'un protocole est l'équation qui lie une valeur publique v_A à sa valeur privée associée s_A, toutes deux liées à l'entité A. On appelle **bi-clé**, ou **paire de clés**, le couple d'éléments (v_A, s_A).

Le **niveau de sécurité** d'un protocole interactif à 3 passes repose sur la longueur du défi d. Elle correspond à la probabilité minimale de fraude d'un tiers à chaque tour d'exécution.

La sécurité d'un protocole interactif s'évalue par sa capacité à sécuriser les 3 modes de connexions possibles : entre prouveur et vérifieur, honnête et/ou malhonnête. La Figure 1.8 permet de visualiser ces connexions : la malhonnêteté d'un individu est symbolisée par un tilde ($\tilde{A}$ et $\tilde{B}$).

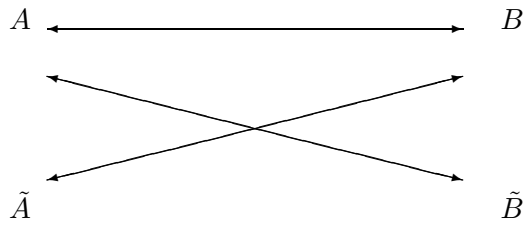


FIG. 1.8 – Sécurité d'un protocole interactif

On s'assure du succès des authentifications quand les deux parties sont honnêtes (A/B), de la protection contre les attaques dans le cas d'une usurpation d'identité ($\tilde{A}/B$), et de la protection du secret ($A/\tilde{B}$).

1.4.2 Protocoles sûrs à divulgation nulle de connaissance

Deux articles majeurs constituent les fondements de leurs preuves de sécurité.

En 1985, Goldwasser, Micali et Rackoff [GMR89] rédigent le premier article, pris encore comme référence 20 ans plus tard [Gol02], qui posent les bases de la mesure de complexité de l'information et de l'estimation de connaissance apportée dans les preuves interactives de connaissance d'un secret. Ils mettent en avant l'interactivité des preuves qui permet de réduire cet apport de connaissance, en particulier avec l'exemple d'un système de preuve interactive de la non-résiduosité quadratique. Le livre de D. Stinson [Sti95] consacre un chapitre à ce concept.

Lors de l'exécution de ces protocoles, l'information transmise par Alice est uniquement la connaissance d'un témoin de la connaissance du secret. Aucune information, même partielle, ne transpire des échanges effectués, car ces protocoles possèdent la propriété d'être simulables par des machines de Turing probabilistes à temps polynomial. Concrètement, ce concept se traduit par la non distinction entre la génération de triplets produits selon un mode public, et celle selon un mode privé : quiconque peut générer des triplets valides, sans connaître le secret. Dans un **mode public**, n'importe qui peut produire des triplets calculés à partir de n'importe quel défi et de n'importe quelle réponse, en utilisant la clé publique. Dans un **mode privé**, celui qui connaît le secret peut produire un aléa, calculer l'engagement correspondant, puis calculer la réponse à n'importe quel défi, en utilisant le secret.

En 1990, Louis Guillou et Jean-Jacques Quisquater [GQ90] ont imaginé l'histoire de la caverne magique afin d'expliquer simplement le concept de preuve sans transfert de connaissance.

Le schéma suivant décrit cette caverne : la caverne possède une entrée E d'où l'on ne peut voir le point de rencontre de 3 couloirs appelé X. Le point M isolé visuellement du point X et du point E, symbolise la porte magique que l'on peut franchir si l'on détient le mot de passe, pour passer d'un couloir à l'autre.

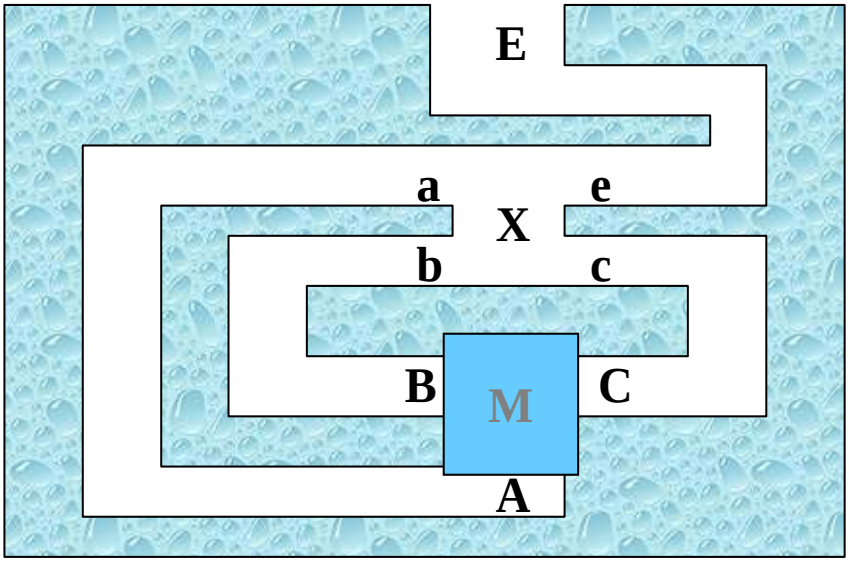


Fig. 1.9 – La caverne magique

Un protocole trivial non zero-knowledge

Alice, détentrice du secret, et Bob, qui va vérifier que Alice connaît ce secret, se positionnent au point X. Alice entre dans le couloir b-B, actionne la porte magique à l'aide du mot de passe et réapparaît au point c, où Bob constate sans aucun doute possible que Alice connaît le secret (le mot de passe). Une ou plusieurs répétitions de cette expérience est une preuve irréfutable de la connaissance du secret par Alice.

Dans ce cas, le protocole ne peut être simulé par une machine qui ne connaît pas le secret.

Un protocole zero-knowledge

Alice et Bob se placent au point E. Alice entre seule dans la caverne jusqu'au point X et choisit un des trois chemins pour arriver au point A, B ou C. Bob rentre à son tour jusqu'au point X et demande à Alice d'apparaître au point a, b ou c, qu'il décide au hasard. Si besoin est, Alice utilise le mot de passe pour franchir la porte magique, et apparaît à l'endroit indiqué.

Les trois notions de base définies par Goldwasser, Micali et Rackoff [GMR89] pour définir la sécurité des protocoles «zero-knowledge » sont vérifiées :

- Toute personne connaissant le secret réussit systématiquement l'expérience.
- L'expérience peut être réitérée et un imposteur n'aura toujours qu'1 chance sur 3 à chaque répétition du protocole pour duper Bob.
- Le protocole ne laisse transpirer aucune information sur le contenu du secret : en effet, personne ne peut avoir la certitude, en étant témoin de la scène de la caverne magique, que les deux personnes du protocole sont *de mèche* ou bien qu'elles sont honnêtes et fidèles au rôle qu'on leur attribue.
  Ainsi la situation où deux personnes ne connaîtraient pas le secret, et simuleraient une série de scènes, dont elles auraient convenu des défis par avance, serait aussi convaincante qu'une situation réelle de dialogue entre un prouveur et un vérifieur. Or une simulation ne transfère évidemment pas le secret (puisqu'il n'est pas connu), donc une authentification réelle réussie ne transfère pas non plus la connaissance du secret.

La sécurité de ce protocole interactif à 3 passes repose sur les trois propriétés suivantes :

- **Complétude** : Si le prouveur connaît la clé privée alors le vérifieur doit l'accepter avec une probabilité écrasante.
- **Significativité** : Si le prouveur ne dispose pas de clé privée alors quelle que soit la stratégie choisie par le prouveur, le vérifieur doit le rejeter avec une probabilité écrasante.
- **Zero-knowledge** : Quelle que soit sa stratégie, le vérifieur n'arrivera jamais à récupérer le secret de la factorisation du module détenu par le prouveur, ni même une partie.

Formellement, les définitions suivantes décrivent un protocole interactif à 3 passes à divulgation nulle de connaissance. Dans un premier temps, une généralisation du Forking Lemma démontré par D. Pointcheval et J. Stern [PS96], est énoncée.

Lemme 1.4.1 *(Forking Lemma) Soit ξ un prédicat à deux variables.*
Soient X et Y deux ensembles finis et $Y1$ un sous-ensemble de Y.
Soient x une variable aléatoire à valeurs dans X, et y, y^* deux variables aléatoires de loi uniforme sur Y. On suppose que x, y, y^* sont indépendantes.
Alors on a l'implication suivante :

$$\forall \epsilon', \theta \in [0,1], Pr[\xi(x,y)] \geq \epsilon' \Rightarrow Pr[\xi(x,y), \xi(x,y^*), y - y^* \notin Y1] \geq \left(\theta - \frac{|Y1|}{|Y|}\right) \theta(\epsilon' - \theta)$$

Preuve. On introduit $B_\theta = \{\alpha \in X / Pr[\xi(x,y)/x = \alpha] > \theta\}$
Dans un premier temps, on calcule $Pr[x \in B_\theta]$ puis on recherche une borne pour la probabilité $Pr[\xi(x,y), \xi(x,y^*), y - y^* \notin Y1]$.

On vérifie que si $Pr[\xi(x,y)] \geq \epsilon'$ alors $Pr[x \in B_\theta] \geq \epsilon' - \theta$.
Développons $Pr[\xi(x,y)]$:

$$Pr[\xi(x,y)] = \sum_{x \in B_\theta} Pr[\xi(x,y)/x = \alpha] Pr[x = \alpha] + \sum_{x \notin B_\theta} Pr[\xi(x,y)/x = \alpha] Pr[x = \alpha]$$

Dans le premier terme de la somme, on a $Pr[\xi(x,y)/x=\alpha] \leq 1$ d'après la définition d'une probabilité, et pour le deuxième terme, on a $Pr[\xi(x,y)/x=\alpha] \leq \theta$, d'après la définition de B_θ . Donc, on a l'inégalité suivante :

$$Pr[x \in B_\theta] + \theta(1 - Pr[x \in B_\theta]) \geq Pr[\xi(x,y)] \geq \epsilon'$$

soit encore :

$$Pr[x \in B_\theta] \geq \frac{\epsilon' - \theta}{1-\theta} \geq \epsilon' - \theta$$

Voici maintenant la preuve du Forking Lemma : on étudie la probabilité $Pr[\xi(x,y), \xi(x,y^*), y - y^* \notin Y1]$:

$$Pr[\xi(x,y),\xi(x,y^*),y-y^* \notin Y1] \geq \sum_{\substack{\alpha,\beta \\ \xi(\alpha,\beta) \\ \alpha \in B_\theta}} Pr[\xi(x,y),\xi(x,y^*),y-y^* \notin Y1/x=\alpha,y=\beta]Pr[x=\alpha,y=\beta]$$

Or, si $\alpha \in B_\theta$, alors en tenant compte des deux inégalités suivantes :
- $Pr[\xi(x,y^*), y^* - \beta \notin Y1/x=\alpha] + Pr[\xi(x,y^*), y^* - \beta \in Y1/x=\alpha] = Pr[\xi(x,y^*)/x=\alpha] \geq \theta$
- $Pr[\xi(x,y^*), y^* - \beta \in Y1/x=\alpha] \leq Pr[y^* - \beta \in Y1] = \frac{|Y1|}{|Y|}$

on en déduit que

$$Pr[\xi(x,y^*), y^* - \beta \notin Y1/x=\alpha] \geq \theta - \frac{|Y1|}{|Y|}$$

En conséquence

$$\begin{aligned}
Pr[\xi(x,y),\xi(x,y^*),y-y^* \notin Y1] &\geq \sum_{\substack{\alpha,\beta \\ \xi(\alpha,\beta) \\ \alpha \in B_\theta}} Pr[\xi(x,y^*),y^*-\beta \notin Y1/x=\alpha,y=\beta]Pr[x=\alpha,y=\beta] \\
&\geq \left(\theta - \frac{|Y1|}{|Y|}\right) \sum_{\substack{\alpha,\beta \\ \alpha \in B_\theta}} Pr[\xi(x,y)/x=\alpha,y=\beta]Pr[x=\alpha,y=\beta] \\
&\geq \left(\theta - \frac{|Y1|}{|Y|}\right) \sum_{\substack{\alpha \\ \alpha \in B_\theta}} Pr[\xi(x,y)/x=\alpha]Pr(x=\alpha) \\
&\geq \left(\theta - \frac{|Y1|}{|Y|}\right) \theta \sum_{\alpha \in B_\theta} Pr[x=\alpha] \\
&\geq \left(\theta - \frac{|Y1|}{|Y|}\right) \theta(\epsilon' - \theta)
\end{aligned}$$

□

Définition 1.4.1 *On appelle* ***protocole interactif à 3 passes*** *le couple* (A,B) *constitué de deux machines de Turing probabilistes à temps polynomial et vérifiant la propriété suivante (Figure 1.10) :*

$$\forall(x,y,w_a,w_b)\ ,\exists(W,d,D,l)\ /$$
$$A_{w_a}(x,d)=(W,D) \quad et \quad B_{w_b}(y,W,D)=(d,l) \quad et \quad l \in \{0,1\}$$

Le triplet (W,d,D) *est unique et se note* $VUE\langle A(x),B(y)\rangle$.

On désigne l *la réponse du protocole (ou* $\langle A(x),B(y)\rangle$*) : si* $\langle A(x),B(y)\rangle = 1$ *(ou noté seulement* $\langle A(x),B(y)\rangle$*), le prouveur* A *est accepté par le vérifieur* B.

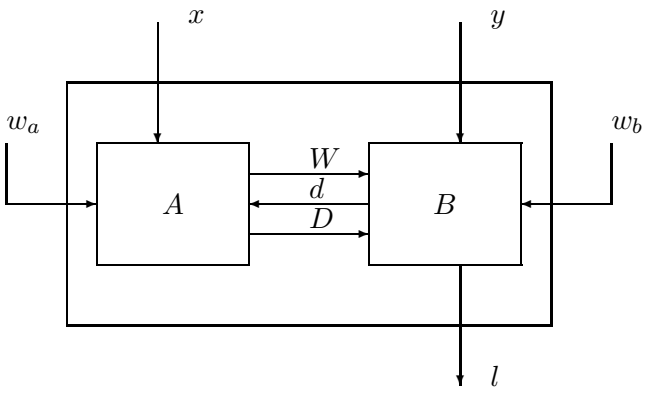


FIG. 1.10 – Protocole interactif à 3 passes

Définition 1.4.2 *Soit F le prédicat associé à la factorisation défini par $F((p_1, p_2), n) = (n = p_1 \times p_2)$.*

On dit qu'un prédicat Ξ est ***équivalent au prédicat*** *F, si la connaissance d'un témoin pour le prédicat Ξ implique la connaissance de la factorisation de n :*

$$\exists M \in MT / \forall (x, y),\ \Xi(x, y) \Rightarrow M(x, y) = (p_1, p_2)$$

Définition 1.4.3 *On dit qu'un protocole interactif (A, B) est une* ***preuve zero-knowledge de connaissance de la factorisation du module*** *si les trois propriétés suivantes sont respectées :*

1. *Le prédicat Ξ associé au protocole est équivalent au prédicat associé à la factorisation du module.*
2. *Propriété de «* ***complétude*** *» (completeness) :*
   Pour tout y valide, si A connaît un témoin x de y pour le prédicat Ξ, alors A convainc B. Formellement :

$$\forall (x, y), \Xi(x, y) \Rightarrow Pr_{w_a, w_b}[\langle A_{w_a}(x), B_{w_b}(y)\rangle] = 1$$

3. *Propriété de «* ***significativité*** *» (soundness) de seuil de sécurité ϵ :*
   Pour tout y valide, un prouveur malhonnête $\tilde{A}$ n'est pas capable de convaincre B avec une probabilité supérieure à ϵ de manière non négligeable s'il ne connaît pas un témoin de y, soit formellement :

$$\forall c > 0, \exists M \in MT / \forall \tilde{A} \in MT, \exists n_0, \forall |y| > n_0,$$

$$Pr[\langle \tilde{A}_{w_a}(y), B_{w_b}(y)\rangle] \geq \epsilon + \frac{1}{|y|^c} \Rightarrow Pr[\Xi(M(y), y)] \geq 1 - e^{-|y|}$$

4. *Propriété de «* ***sans connaissance*** *» (zero-knowledge) :*
   Pour tout y valide, toute information obtenue par un vérifieur $\tilde{B}$ au cours de l'exécution du protocole avec A, peut être obtenue par $\tilde{B}$ à l'aide d'une simulation sans interaction avec A, soit formellement :

$$\forall \alpha > 0, \exists M \in MT, \forall \tilde{B} \in MT,\ \forall (x,y), \Xi(x,y)$$

$$\sum_{(x_1,y_1,z_1)} |Pr_{w_a,w_{\tilde{b}}}[VUE\langle A_{w_a}(x), \tilde{B}_{w_{\tilde{b}}}(y)\rangle = (x_1,y_1,z_1)] - Pr_{w_m}[M_{w_m}(y) = (x_1,y_1,z_1)]| < \frac{1}{|y|^\alpha}$$

Le **seuil de sécurité** de la preuve de connaissance de la factorisation, noté ϵ, est la probabilité maximale d'usurpation d'identité : tout prouveur qui s'authentifie avec une probabilité supérieure à ϵ de manière non négligeable, connaît nécessairement la factorisation du module.

1.4.3 Construction de ces protocoles d'authentification

La construction de nouveaux schémas d'authentification 3 passes se déduit pour la grande majorité de ceux déjà existants. Nous regardons en particulier les protocoles basés sur la factorisation du module et détaillons l'historique des articles sur les schémas dits de type Fiat-Shamir. La déclinaison des protocoles basés sur le calcul du logarithme discret, en particulier GPS, sera rappelée.

Modes et procédés de composition

Dans la littérature, il existe 3 modes différents pour un schéma de sécurité donné : les modes séquentiel et parallèle du schéma d'authentification, et son mode signature. Les deux premiers modes sont des techniques utilisées pour diminuer la probabilité de fraude, sans modifier les paramètres des schémas : le **mode séquentiel** consiste à réexécuter le schéma un nombre de fois donné, de manière consécutive, et le **mode parallèle** consiste à exécuter le schéma un nombre de fois donné, de manière simultanée.

La structure même des schémas peut se déduire à travers plusieurs déclinaisons possibles : le passage d'une à plusieurs valeurs secrètes par tour d'exécution (Fiat-Shamir → Feige-Fiat-Shamir), la modification de l'exposant public (GQ → Ong-Schnorr), l'utilisation de petites valeurs publiques (Ong-Schnorr → GQ2) ainsi que les modifications opérées afin de permettre l'élaboration des schémas dits, à témoins indistinguables (GQ → GQ modifié [Oka92], Schnorr → Schnorr modifié [Oka92], Ong-Schnorr → Ong-Schnorr modifié[FF02]). Des morphismes généraux applicables à tous les protocoles d'authentification basés sur la factorisation ou le logarithme discret, comme l'article de Shamir [Sha84] publié dès 1984, ont été mis en évidence en 2004 par Bellare et al. [BNN04].

Dans le paragraphe suivant, nous nous intéressons aux schémas de type Fiat-Shamir, en reprenant les mêmes notations afin de mettre en évidence ses déclinaisons.

Protocoles interactifs du type Fiat-Shamir

Dès 1987, une application des preuves d'apport de connaissance nul est publiée dans l'article de Feige-Fiat-Shamir [FFS89]. Ce dernier généralise le schéma de Fiat-Shamir inventé un an plus tôt, à une déclinaison pour des secrets multiples, et apporte la preuve de sécurité de ce protocole, qui sera le premier d'une longue série de protocoles

basés sur le problème difficile du calcul de racines carrées modulo un grand nombre composé.

La structure de ces protocoles est présentée grâce à la Figure 1.11, en utilisant les notations suivantes : soient v et s les exposants public et privé tels que $v.s = 1 \bmod \phi(n)$, v_A et s_A les valeurs publique et privée relatives à Alice, r une valeur aléatoire et k et m deux paramètres. Dans le cas de Fiat-Shamir, $k = 1$ et $v = 2$.

Remarque : Deux types d'équation générique peuvent être utilisés de manière équivalente. Les termes de **indirecte** ($v_{A_i} s_{A_i}^v = 1 \bmod n$) et **directe** ($v_{A_i} = s_{A_i}^v \bmod n$) désignent ces équations : une multiplication s'opère lors de l'opération de vérification dans le premier cas ($D^v v_{A_1}^{d_1} ... v_{A_m}^{d_m}$), une division dans le deuxième cas ($D^v / (v_{A_1}^{d_1} ... v_{A_m}^{d_m})$).

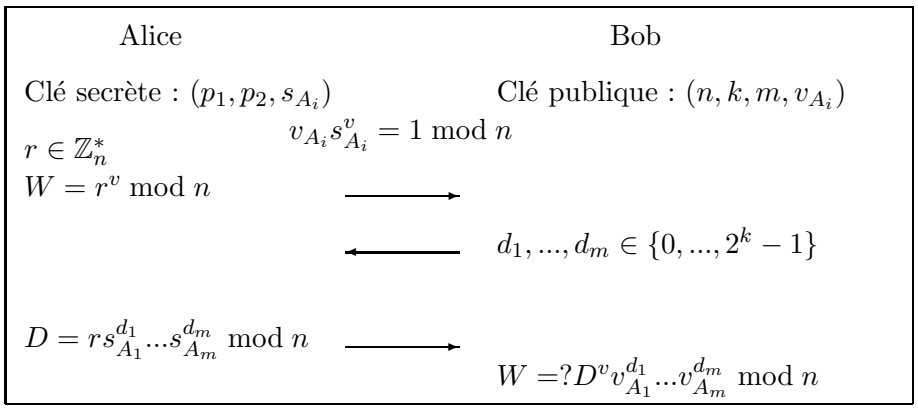


FIG. 1.11 – Schéma général du protocole de type Fiat-Shamir

Le schéma de Fiat-Shamir [FS87] nécessite d'être réitéré plusieurs fois et demande un volume d'informations échangées important ; malgré sa généralisation moins gourmande de l'article de [FFS89], le problème de l'application réelle de ce protocole demeure. Le schéma de Micali-Shamir [MS88] a permis de réduire la complexité du vérifieur en considérant comme petites valeurs publiques les premiers nombres premiers. Seul le module sera donc transmis. Cette technique de l'utilisation de valeurs publiques sera réutilisée par la suite.

En 1988, L. Guillou et JJ. Quisquater [GQ88b][GQ88a][BP02] présentent un nouveau protocole appelé GQ ou GQ1, qui permet de réduire le nombre d'itérations du protocole à une seule. Le niveau de sécurité est fixé en un seul tour d'exécution grâce à une longueur du défi de la taille de la clé. Le prouveur n'a besoin de stocker qu'une seule valeur d'authentification. On observe un gain de temps en transmission et en mémoire, mais le temps de calcul reste important pour une intégration sur des composants à faible puissance de calculs comme les cartes à puce. Ce protocole est utilisé aujourd'hui à grande échelle dans l'authentification client-serveur dans le produit Netware de Novell (NDS).

La même année, un nouveau schéma de type Fiat-Shamir présente l'utilisation d'un exposant public quelconque [OO88]. Le niveau de sécurité n'est donc plus seulement

lié à la place mémoire utilisée, au volume d'informations échangées, mais également à la valeur de l'exposant public. Cependant, ce protocole zero-knowledge demeure plus lent que Fiat-Shamir.

En 1990, H. Ong et C.P. Schnorr [OS91] utilisent l'exposant public v de la forme 2^k, toujours dans une déclinaison des protocoles de type Fiat-Shamir. La preuve de sécurité est basée sur le problème difficile du calcul des racines carrées successives. Les coûts de calculs se voient largement diminués par rapport au schéma RSA. En 1996, V. Shoup [Sho96] donne une démonstration de sécurité contre les attaques basées sur la difficulté à calculer les racines carrées successives. Il démontre que le schéma de Ong-Schnorr est sûr mais pas zero-knowledge. C.P. Schnorr [Sch97] reprend également les preuves de sécurité de ce protocole, avec un module quelconque.

En 1999, un rapport technique est rédigé [GQU01] décrivant le protocole le plus performant de type Fiat-Shamir, appelé GQ2. C'est un protocole crée pour conserver les avantages du premier schéma de leurs inventeurs, GQ, et pour être utilisable dans des conditions de ressources réduites en utilisant les atouts des schémas de type Fiat-Shamir, comme l'exposant public de la forme une puissance de 2, et l'utilisation de petites valeurs publiques. Un autre avantage de ce protocole est sa sécurité prouvée équivalente à la factorisation du module sous certaines conditions, grâce à la nature de ces nombres publics : des carrés de petits nombres premiers. En 2004, une généralisation de ce schéma a été apportée, afin de montrer la compatibilité de la génération des clés de ce protocole avec les modules RSA existants.

Le Tableau 1.1 donne un récapitulatif de ces protocoles de type Fiat-Shamir en fonction de leurs caractéristiques.

	Exposant(v)				Multiplicité (m)		v_A	
	2	2^k	premier	qcq	unique	multiple	petits	qcq
Fiat Shamir ([FS87])	⋆				⋆			⋆
Feige Fiat Shamir ([FFS89])	⋆					⋆		⋆
Micali Shamir ([MS88])	⋆					⋆	⋆	
GQ ([GQ88b])			⋆		⋆	⋆		⋆
Otha-Okamoto ([OO88])				⋆	⋆	⋆		⋆
Ong-Schnorr ([OS91])		⋆			⋆	⋆		⋆
GQ2 ([GQU01])		⋆			⋆	⋆	⋆	

TAB. 1.1 – Protocoles du type Fiat-Shamir

Les preuves de sécurité des protocoles basés sur la difficulté à calculer les racines carrées successives ont été traitées par étapes successives durant les 15 dernières années, en fonction des cas considérés : sécurité équivalente au calcul des racines carrées, sécurité équivalente à la factorisation du module dans le cas des entiers de Blum et dans le cas d'entiers composés quelconques. Ces preuves se trouvent dans les articles [OS91] [Mic94] [Sho96] [Sch97].

Autres protocoles interactifs

Une autre ligne de protocoles d'authentification de type Chaum-Evertse-Graaf repose sur le problème difficile du logarithme discret : Chaum-Evertse-Graaf [CEvdG87],

Schnorr [Sch89], GPS [Gir01][PS98] et Poupard-Stern [PS00].

[LOG] : Etant donné un entier p premier, un générateur g de $\mathbb{Z}_p^*$ ainsi qu'un élément I de $\mathbb{Z}_p^*$, trouver l'entier S de $\mathbb{Z}_{p-1}^*$ tel que $I = g^S \bmod p$.

Le protocole GPS, inventé par M. Girault, G. Poupard et J. Stern repose à la fois sur le problème difficile du logarithme discret et sur le problème de la factorisation des grands nombres. Il fait partie des protocoles dits à très bas-côuts, qui moyennant des précalculs, permettent une authentification sans l'utilisation de micro-processeur. Les RFID tags sont des composants qui utilisent ce protocole, où seuls des calculs sur portes logiques sont nécessaires.

Chapitre 2

Performance et sécurité du protocole GQ2

Le protocole GQ2 appartient à la famille des protocoles interactifs à 3 passes à divulgation nulle de connaissance de type Fiat-Shamir. Ce chapitre est issu d'un travail de collaboration avec Francois Daudé et Louis Guillou et a donné lieu aux articles [BDG04a] et [BDG04b].

Le contexte d'élaboration du protocole GQ2 est important à souligner. Il est né, en 1999, d'un besoin exprimé par le groupement des cartes bancaires (GIE). A cette époque, la communauté scientifique est mise à contribution pour réfléchir à une solution de sécurité concernant l'évolution des cartes bancaires du parc français.

Pour répondre à cette attente, L. Guillou et JJ. Quisquater proposent un protocole, évolution de leur premier protocole GQ1. Les performances de ce nouveau schéma à sécurité élevée, sont particulièrement indiquées pour les contraintes d'intégration sur cartes à puce [BDG04a]. La comparaison du nombre de calculs nécessaires entre RSA et GQ2 dans leur version dynamique, ne laisse aucun doute quant à l'efficacité de GQ2. De plus, grâce à la liberté proposée sur les paramètres, l'utilisateur peut faire varier ces derniers pour s'adapter au contexte d'application.

Nous réservons une grande part de ce chapitre à l'analyse complète de sécurité de ce protocole [BDG04b] afin de conclure sur les conditions que doivent respecter les paramètres GQ2 pour être une preuve de connaissance de la factorisation du module.

Dans la suite, nous présenterons une version généralisée du protocole GQ2 qui répond en plus à un besoin de compatibilité avec les services utilisant déjà un module RSA. A l'aide d'un paramètre d'adaptation, nous générerons des clés GQ2 à partir d'une clé RSA quelconque, et étendrons cette compatibilité dans le cas de modules multifacteurs.

2.1 Origine : la sécurité des cartes bancaires

Les cartes à puce permettent de mettre en oeuvre un contrôle d'accès sécurisé. Ce système est largement utilisé dans le domaine de la télévision numérique ou dans celui

des téléphones portables.

Le marché des cartes bancaires en France s'est également rapidement intéressé à ce composant de sécurité au début des années 80. Les contraintes d'intégration sont cependant grandes : le volume de données stockables est limité, et le nombre d'opérations élémentaires doit correspondre à la puissance de calculs limitée du microprocesseur. D'un simple objet de stockage de données aujourd'hui, la carte bleue pourrait devenir un véritable acteur de la sécurité de transactions bancaires.

2.1.1 Identification et authentification

Dans la sécurité des cartes bancaires, il est important de distinguer les phases d'identification et d'authentification.

L'**identification** d'un élément est le procédé qui consiste à reconnaître un élément comme appartenant à un ensemble. Un cas concret : le numéro de sécurité sociale est propre à chaque assuré social. Il caractérise à lui seul sa correspondance avec une personne unique.

L'**authentification** d'un élément est le procédé qui consiste à pouvoir s'assurer l'appartenance de cet élément à un ensemble. Un cas concret : les cartes téléphoniques non nominatives, les télécartes. France Télécom doit s'assurer que les cartes utilisées dans les cabines téléphoniques, proviennent de sa production.

2.1.2 Cartes à puce contre les cartes magnétiques

Les cartes magnétiques sont encore majoritairement répandues dans de grands pays industriels comme les Etats-Unis et le Japon. Ces cartes assurent cependant seulement la partie authentification de la carte avec sa banque, via des données identifiantes du propriétaire de la carte : la banque identifie et s'assure que le compte qui sera débité, existe. Le terminal n'est pas utilisé comme acteur de la sécurité. L'article de vulgarisation [Pat02] relatent les étapes d'une transaction bancaire.

L'introduction de la carte à puce a permis d'augmenter la sécurité de la phase d'identification. En effet, elle crée un lien supplémentaire entre la carte et le propriétaire de la carte, grâce au code PIN : l'utilisateur de la carte est le propriétaire de la carte, et donc du compte à débiter.

Parallèlement, la partie authentification se voit alors allégée, en élevant le pouvoir du terminal qui peut authentifier cette carte comme appartenant au parc légal bancaire, via une signature RSA (donnée statique liée à la carte). Le terminal commence à être utilisé comme un acteur de la sécurité. Il effectue la vérification de la signature RSA, envoyée par la carte. Souvent, au-dessus d'un certain montant, la sécurité s'élève, et dépasse de manière évidente celle des cartes magnétiques en établissant également la liaison avec la banque, via une authentification DES, afin de s'assurer de l'existence du compte à la banque par les données identifiantes émises. Le code PIN intervient également dans cette authentification.

Cependant, si la phase d'identification par l'entrée du code PIN est contournée, et la liaison avec la banque ne se fait pas, la sécurité repose uniquement sur la signature RSA. Ainsi, en 1998, Humpich factorise le module commun à toutes les cartes du parc, pourtant annoncé comme obsolète par la communauté scientifique (RSA-320), puis élabore et intègre de fausses Valeurs de Signature (VS) à partir de données identifiantes factices... qui ne sont pas vérifiées par la banque pour de petits montants.

En 1999, la réaction du GIE des cartes bancaires a été d'augmenter la taille du module de 320 bits (choisi en 1982) à 768 bits tout en gardant le principe de l'authentification statique. Dans le cas d'une authentification statique, un composant bon marché suffit ; l'ajout d'un crypto processeur afin de supporter des calculs plus importants est indispensable pour passer à une authentification dynamique RSA-768 ou RSA-1024, à temps d'authentification carte raisonnable ($<$ 1 seconde). Mais cette décision ne fait que repousser le risque de cartes «clones », car comme tout système cryptographique digne de ce nom, la taille des clés doit évidemment évoluer en même temps que la puissance des ordinateurs. Pour l'instant le choix des banques est de rembourser systématiquement en cas de fraude, plutôt que de repenser le système de sécurité des cartes à puce en profondeur.

2.1.3 De l'authentification statique à l'authentification dynamique

Dans l'attente du renouvellement total du parc des cartes bancaires, et pour parer à une future attaque inévitable de «yes-cards-768» dans le futur, un processus systématique «on line » (liaison directe avec la banque) pour tout paiement automatique sans présence humaine a été mis en place : dans les stations service 24-24, un message «Appel en cours» d'une durée de près de 30 secondes s'affiche le temps de l'appel à la banque.

Une réponse possible à cette attaque consiste à augmenter encore davantage le rôle joué par le terminal dans la sécurité des transactions. En effet, à l'aide d'une authentification dynamique, il n'est plus possible de calculer par avance, et de stocker dans la carte, une valeur statique comme la Valeur de Signature précédente. De ce fait les vraies fausses cartes précédentes, appelées «yes-cards » ne peuvent plus être utilisées. La carte doit envoyer le certificat de son module propre lors de sa connexion avec le terminal, et prouver sa connaissance du secret (c'est à dire dans le cas de GQ2, la factorisation du module, et l'exposant privé RSA dans le cas de RSA). Pour cela, une authentification dynamique, parmi celles spécifiées dans la norme ISO/IEC 9798-5, prouvées à divulgation nulle de connaissance, peut être envisagée.

Aujourd'hui, le composant de sécurité des cartes bancaires, la puce électronique, n'est pas utilisé pour sécuriser les transactions sur Internet : ces protocoles pourraient être une solution de sécurité pour les services d'achats en ligne.

2.1.4 Authentification dynamique GQ2 dans les cartes à puce

Elaboré en 1999 par Louis Guillou et Jean-Jacques Quisquater, le protocole GQ2 était destiné à répondre au problème posé par le GIE cartes bancaires, d'augmenter la sécurité des cartes bancaires en maintenant les performances, à moindre coût. A l'époque, le choix s'est finalement tourné vers une conservation de la formule de l'authentification statique RSA en augmentant la longueur des clés.

Le choix aujourd'hui d'une authentification dynamique à moyen terme semble s'affirmer. RSA demeure trop souvent l'unique solution à considérer. L'intégration d'un crypto-processeur (silicium supplémentaire à rajouter dans la carte à puce afin d'accélérer les calculs arithmétiques de base) serait cependant indispensable afin de conserver un temps d'authentification aux alentours de la seconde. La sécurité de RSA éprouvée depuis des années, sa simplicité de construction, ainsi que le statut d'algorithme public depuis 1998, continuent à mener la vie dure aux autres solutions de

sécurité proposées depuis.

Parmi les solutions proposées, les courbes elliptiques ont permis de réduire la taille des clés tout en maintenant la sécurité. Un engouement important de ces nouveaux protocoles de sécurité dans la communauté scientifique n'a cependant pas eu l'effet escompté du coté des investisseurs qui demeurent frileux à être les premiers utilisant cette technique dans leur système de sécurité. De plus, il semble que l'utilisation d'un crypto-processeur soit également nécessaire pour mettre en oeuvre cette solution dans les cartes à puce.

Le protocole GQ2 conserve le lien avec le problème difficile mathématique utilisé pour la sécurité RSA : la factorisation des grands nombres. Ce point commun sur la sécurité s'accompagne également d'un déploiement analogue lors de la phase génération de clés. En 2004, un nouveau brevet est rédigé par les inventeurs de GQ2, pour mettre en avant la compatibilité du protocole avec la PKI RSA : on peut générer des clés GQ2 à partir de n'importe quelles clés RSA existantes.

Suite au premier brevet de ce protocole de 1999, deux rapports techniques [GQU01][GQ01] ont été rédigés. En 2004, deux nouveaux articles reprennent et détaillent la preuve de sécurité [BDG04b] et les performances de ce protocole [BDG04a].

2.2 Spécifications du protocole

Cette partie présente les étapes de l'intégration du protocole GQ2 : de la construction des clés privées à partir de petits nombres de base publics aux calculs et échanges de données lors de l'exécution du protocole.

2.2.1 Elaboration des bi-clés à partir de deux facteurs congrus à 3 modulo 4

La description suivante est un cas particulier des spécifications de GQ2 présentes dans la norme ISO/IEC 9798-5 [ISO04] : les 2 nombres premiers p_1 et p_2 sont congrus à 3 modulo 4.

GQ2 fait appel aux trois paramètres suivants :
- k : **paramètre de sécurité**.
- m : **paramètre de multiplicité**.
- v : **exposant de vérification** et défini par $v = 2^{k+1}$.

La construction d'une bi-clé GQ2 s'effectue selon les étapes suivantes :
- On choisit aléatoirement deux nombres premiers p_1 et p_2 congrus à 3 modulo 4
- On calcule le module n égal au produit de p_1 par p_2
- La clé publique GQ2 se compose du module n et de m nombres publics notés $(G_1, ..., G_m)$, chaque G_i étant le carré d'un petit nombre premier noté g_i et appelé **nombre de base**. : $\forall i \in \{1, ..., m\},\ G_i = g_i^2$
- La clé publique GQ2 doit vérifier la propriété suivante :
  Pour au moins un nombre de base noté g, nous avons : $(g|p_1) = -(g|p_2)$
- La clé privée GQ2 se compose des nombres premiers p_1, p_2 et de m nombres secrets notés $(Q_1, ..., Q_m)$ reliés aux nombres publics par les équations génériques suivantes :

$$\forall i \in \{1, ..., m\}, G_i Q_i^v = 1 \bmod n$$

2.2.2 Génération des clés privées

Proposition 2.2.1 *Soient p_1, p_2 deux nombres premiers congrus à 3 modulo 4 qui composent le module n. Chaque facteur s'écrit sous la forme $p_i = 2q_i + 1$ avec q_i impair.*

Soit l'équation générique indirecte $GQ^v = 1 \bmod n$.

La valeur privée relative au nombre de base g dans le groupe multiplicatif de p_i se déduit par le calcul de $Q_i = g^{-2.x_i} \bmod p_i$, où

$$x_i = ((p_i + 1)/4)^{k+1} \bmod (p_i - 1)/2$$

On en déduit la valeur privée relative au nombre de base g dans le groupe $\mathbb{Z}_n^*$ par $Q = CRT(Q_i, i = 1, 2)$.

D'une autre manière, en une seule étape, on peut calculer $Q = g^{-2.y} \bmod n$ avec

$$y = (q_1 q_2 + 1)/2^{k+1} \bmod (p_1 - 1)(p_2 - 1)/4$$

Preuve. Il suffit de vérifier l'équation générique $GQ^v = 1 \bmod n$ (cf la preuve plus générale de la Proposition 3.2.1).

□

2.2.3 Le protocole d'authentification

Le protocole GQ2 s'effectue entre un prouveur et un vérifieur. Le vérifieur connaît la clé publique GQ2 $(n, G_1, ..., G_m)$ et le prouveur connaît la clé privée GQ2 $(p_1, p_2, Q_1, ..., Q_m)$. Ils possèdent en commun l'exposant de vérification $v = 2^{k+1}$ et le paramètre de multiplicité m.

Alice		Bob
Clé privée : (p_1, p_2, Q_i)		Clé publique : (n, k, m, g_i)
$r \in \mathbb{Z}_n^*$ $W = r^{2^{k+1}} \bmod n$	$\longrightarrow$	
	$\longleftarrow$	$d_1, ..., d_m \in \{0, ..., 2^k - 1\}$
$D = rQ_1^{d_1}...Q_m^{d_m} \bmod n$	$\longrightarrow$	
		$W = ?D^v(g_1^2)^{d_1}...(g_m^2)^{d_m} \bmod n$

FIG. 2.1 – Schéma du protocole GQ2 pour $v = 2^{k+1}$

Le prouveur GQ2 réalise alors systématiquement les étapes suivantes (cf Figure 2.1) :

1. Sélection d'un nombre aléatoire positif et inférieur à n, noté r
2. Calcul de $W = r^v \bmod n$ appelé **engagement** et noté W

3. En réponse à un défi émis par le vérifieur, consistant en m nombres aléatoires de k-bits notés $(d_1, ..., d_m)$, calcul du nombre $D = r.Q_1^{d_1}...Q_m^{d_m} \bmod n$ appelé **réponse** et noté D
4. Effacement du nombre aléatoire r.

Le vérifieur réalise systématiquement les étapes suivantes :

1. Réception de l'engagement W
2. Sélection de m nombres aléatoires de k-bits notés $(d_1, ..., d_m)$
3. En réponse à un nombre D émis par le prouveur, calcul du nombre $W' = D^v.G_1^{d_1}...G_m^{d_m} \bmod n$ et vérification de la condition $W' = W$
4. Si la condition précédente est vérifiée, acceptation du prouveur.

Lorsque ce protocole est effectué un nombre t de fois, on parle du protocole **GQ2 itéré à l'ordre** t.

Remarque : En notant, $G = (n, g_1, ..., g_m)$, $Q = (p_1, p_2, Q_1, ..., Q_m)$ et $d = (d_1, ..., d_m)$ on peut définir les deux machines de Turing à temps polynomial :
– $A_r(Q, d) = (W, D)$ où $W = r^v \bmod n$ et $D = r.Q_1^{d_1}...Q_m^{d_m} \bmod n$
– $B_d(G, W, D) = (d, l)$ où $l = 1 \Leftrightarrow W = D^v.g_1^{2\times d_1}...g_m^{2\times d_m} \bmod n$
GQ2 est bien un protocole à 3 passes au sens de la Définition 1.4.1.

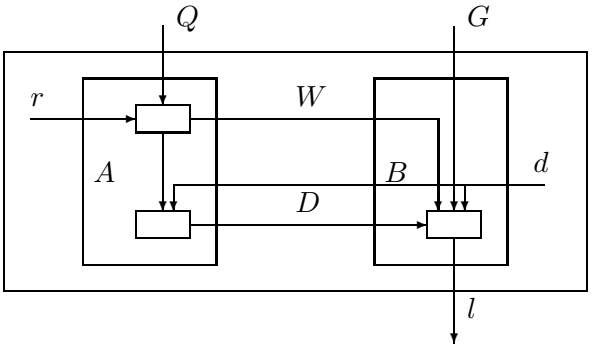


FIG. 2.2 – Schéma du protocole interactif à 3 passes GQ2

2.3 Expérimentation et comparaison des performances GQ2

En termes de performance, une comparaison du protocole d'authentification GQ2 avec RSA en mode dynamique fut effectuée dans le cadre d'une expérience sur carte à puce, avec une clé publique de 1024 bits. De manière plus complète, les résultats sur les complexités de calculs avec les protocoles d'authentification à divulgation nulle de connaissance de la norme ISO/IEC 9798-5 sont présentés.

2.3.1 Performances RSA/GQ2

En 2003, l'intégration des deux protocoles RSA et GQ2 dans une carte bas-coût avec des clés de 1024 bits, rendait compte du rapport de performance RSA/GQ2 : **40**. L'authentification GQ2 aux alentours de la seconde ne nécessite pas l'utilisation de crypto-processeur dans la carte ($k = 8, m = 2, f = 2$).

Le protocole dynamique RSA à 2 passes intégré était le suivant :

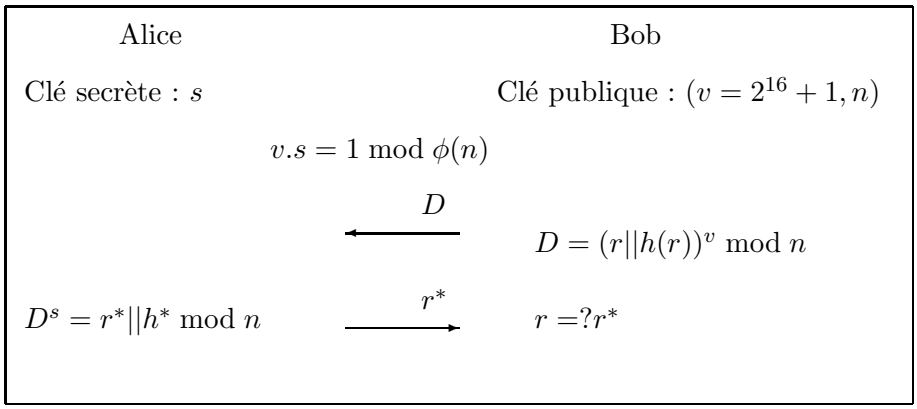


FIG. 2.3 – Protocole d'authentification RSA

Dans le futur, l'augmentation inéluctable des clés avec les performances des ordinateurs conduira à augmenter ce ratio de manière linéaire : la complexité GQ2 augmente comme le carré de la longueur des clés, alors que la complexité RSA augmente comme le cube.

Le Tableau 2.1 détaille les calculs de complexité des protocoles RSA et GQ2 en fonction de la longueur du module n et du nombre de facteurs f qui composent le module.

Voici les estimations faites dans la norme ISO/IEC 9798-5 :

- L'exposant public RSA est de la forme $v = 2^l + 1$
- On raisonne sur le nombre de multiplications modulaires (MM) en se basant sur l'estimation suivante : 1 $CM \Leftrightarrow 0,75$ MM où CM est le nombre de carrés modulaires
- L'algorithme «Square and multiply » : une exponentiation modulaire d'un nombre a de $|n|$ bits à la puissance b, de même taille, coûte $|n|$ CM + $\frac{|n|}{2}$ MM (lié à la probabilité moyenne d'avoir $\frac{|n|}{2}$ bits à 1 en exposant)
- La technique des restes chinois :
  - f exponentiations à coût réduit (de moitié pour deux facteurs, du tiers pour trois facteurs...)
  - les multiplications modulaires à coût réduit impliquent une division par f^2 du coût total des multiplications modulaires.
  - les opérations supplémentaires, notées X, de la technique des restes chinois (les opérations d'addition et de soustraction sont négligés) sont $X = 1.5$ pour

		sans CRT	avec CRT
RSA	$d^s \bmod n$		
	CM	$\|n\|$	$(f.\frac{\|n\|}{f})/f^2$
	MM	$\frac{\|n\|}{2}$	$(X + f.\frac{\|n\|}{2.f})/f^2$
	Total(MM)(A)	$1,25. \mid n \mid$	$(X + \frac{5.\|n\|}{4})/f^2$
	$r^v \bmod n$		
	CM	$\mid v \mid$	
	MM	1	
	Total (MM)(B)	$0,75. \mid v \mid +1$	
GQ2	$r^v \bmod n$		
	CM	$k+1$	$f.(k+1)/f^2$
	MM	0	X/f^2
	$D = r.Q_1^{d_1}.Q_2^{d_2} \bmod n$		
	CM	$k-1$	$f.(k-1)/f^2$
	MM	$\frac{k.m}{2}$	$(X + f.\frac{k.m}{2})/f^2$
	Total (MM)(A)	$k.(m+3)/2$	$(2.X+f.k.(m+3)/2)/f^2$
	$D^v.G_1^{d_1}.G_2^{d_2} \bmod n$		
	CM	$k+1$	
	MM	0	
	Total (MM)(B)	$0,75(k+1)$	

Tab. 2.1 – Comparaison RSA/GQ2

$f = 2$, $X = 4.5$ pour $f = 3$, $X = 9.5$ pour $f = 4$ et $X = 16.5$ pour $f = 5$.

Remarque : voici les étapes de la technique des restes chinois

Soient $p_1, p_2 \in \mathbb{P}$. On précalcule $cr_1 = 1/p_2 \bmod p_1$.
Entrée : W_1, W_2 tels que $W_1 = W \bmod p_1$, $W_2 = W \bmod p_2$
Sortie : W
- $y = W_1 - W_2 \bmod p_1$
- $z = y.cr_1 \bmod p_1$
- $W = z.p_2 + W_2$

2.3.2 Complexités des protocoles de la norme ISO/IEC 9798-5

Dans ce paragraphe, nous comparons les protocoles d'authentification dynamique répertoriés dans la norme ISO/IEC 9798-5 [ISO04] publiée en décembre 2004. Le niveau de sécurité considéré est le même pour chacun des protocoles (2^{-16}), et la longueur du module est fixée à 1024 bits. (cf Tableau 2.2)

On observe la complexité de calculs du côté prouveur (CPC), la complexité de calculs du côté vérifieur (CPV), la capacité de stockage nécessaire au prouveur (CS) et la complexité de communication entre le prouveur et le vérifieur (CM).

Les complexités CPC et CPV sont les plus critiques, car elles jouent un grand rôle dans le temps d'exécution de l'authentification. En particulier, CPC doit être le plus petit possible lorsqu'on est dans un contexte de puissance de calculs très limitée.

	CS(Kbits)	CPC(MM)	CPV(MM)	CM(Kbits)
Fiat Shamir ([FS87])	5,00	11,00	11,00	8,00
GQ1 ([GQ88b])	2,00	33,50	21,50	2,00
GQ2 ([GQU01])	5,50	7,75	3,75	2,00
Schnorr [Sch89])	2,31	200,00	208,00	1,17
GPS ([PS98])	1,16	192,00	200,00	1,27
RSA ([RSA78])	2,50	320,00	13,00	1,84

TAB. 2.2 – Résultats norme ISO/IEC 9798-5

On constate que GQ2, Fiat-Shamir et GQ1 sont les 3 protocoles les moins gourmands en nombres de multiplications modulaires pour CPC et CPV. Les capacités de stockage nécessaires sont du même ordre pour GQ2 et Fiat Shamir mais supérieures à celles des autres protocoles : cette mémoire est utilisée pour stocker les clés privées. Quant au volume de données échangées entre les deux entités, il est très important pour le schéma de Fiat-Shamir.

Il est intéressant de souligner l'équilibre établi entre le prouveur et le vérifieur en termes de complexités de calcul, dans le cadre du protocole GQ2. Ces performances n'entraînent pas des complexités de stockage et de communication pénalisant dans un contexte réel. Ainsi, il offre, par ses caractéristiques, de très bonnes performances en temps d'exécution, à niveau de sécurité élevé.

En effet, grâce à son exposant public de la forme $v = 2^{k+1}$, les complexités de calcul sont réduites pour chaque entité. Le prouveur réduit également ses calculs grâce à la petite taille du défi (km bits où $km < 40$) placé en exposant lors du calcul de la réponse, et les petits nombres de base sont favorables aux performances de la phase de vérification.

Plusieurs paramètres peuvent être ajustés en fonction de la sécurité (longueur du défi et nombre d'itérations t du protocole) et des priorités recherchées dans les applications spécifiques (gain de mémoire, performances de la carte, rapidité de calculs pour le vérifieur). Le paramètre f est le nombre de facteurs du module, et joue un rôle dans l'amélioration des performances côté prouveur grâce à la technique des restes chinois.

	Sécurité	Performance	Stockage mémoire
$k \nearrow$	$\nearrow$	$\searrow$	-
$m \nearrow$	$\nearrow$	-	$\nearrow$
$t \nearrow$	$\nearrow$	$\searrow$	-
$f \nearrow$	-	$\nearrow$	$\nearrow$-

TAB. 2.3 – Ajustement des paramètres de GQ2 aux besoins du service

La carte à puce est l'exemple d'utilisation où GQ2 possède le plus d'avantages en

termes d'intégration par rapport aux autres protocoles. A ses performances à sécurité élevée sur une carte bas-coût s'additionnent également des caractéristiques propres, qui permettent de parer aux 2 attaques physiques classiques : DPA et DFA.

Dans le cas de GQ2, l'attaque DPA ne peut exister puisque l'exposant lors de l'exponentiation modulaire n'est pas secret mais public. De même, pour l'attaque DFA, l'envoi du haché de l'engagement permet de contrer cette attaque.

2.4 Analyse de sécurité du protocole

L'analyse de sécurité du protocole GQ2 s'appuie sur la Définition 1.4.3 : les propriétés «completeness », «soundness » et «zero-knowledge » permettent de valider la sécurité des protocoles interactifs à 3 passes à divulgation nulle de connaissance.

De manière informelle, les propriétés «completeness » et «soundness » permettent de s'assurer que quiconque réussit l'authentification avec une probabilité supérieure à une probabilité ϵ de manière non négligeable, connaît la factorisation du module. En effet, on montre que, sous ces hypothèses, quiconque peut générer des triplets entrelacés valides avec une probabilité non négligeable.

Le caractère zero-knowledge est une propriété supplémentaire qui permet de s'assurer que n'importe quel observateur ne pourra récupérer d'information sur le secret.

Théorème 2.4.1 *Le protocole GQ2 est une preuve zero-knowledge de connaissance de la factorisation du module si* $\epsilon = 1/2^{km}$ *où* ϵ *est le seuil de sécurité de la preuve de connaissance de la factorisation, et si* 2^{km} *croît comme un polynôme en* $|n|$.

2.4.1 Le prédicat associé à GQ2 est équivalent au prédicat associé à la factorisation

Lemme 2.4.1 *Le prédicat associé au protocole GQ2* $\Xi(Q, G) = (\wedge_{i:1..m} G_i Q_i^v = 1 \bmod n)$ *est équivalent au prédicat associé à la factorisation* $F((p_1, p_2), n) = (n = p_1 \times p_2)$.

Preuve. On applique la définition de l'équivalence d'un prédicat avec le prédicat de la factorisation (cf Définition 1.4.2).

La connaissance de G et de la factorisation du module n permet de déterminer le témoin Q de G (cf Proposition 2.2.1). Réciproquement, sachant qu'il existe un nombre de base g tel que $(g|p_1) = -(g|p_2)$ en posant $X = g$ et $Y = Q^{-v/2} \bmod n$, on vérifie que $X^2 = Y^2 \bmod n$ mais que $X \neq \pm Y \bmod n$. On en déduit donc la factorisation de $n = pgcd(X - Y, n) \times pgcd(X + Y, n)$.

□

De fait, si GQ2 est une preuve zero-knowledge de la connaissance du prédicat associé à GQ2, elle est aussi une preuve zero-knowledge de la connaissance de la factorisation du module n.

2.4.2 GQ2 respecte la propriété «completeness »

Proposition 2.4.1 *Le protocole GQ2 est un protocole interactif à 3 passes qui vérifie la propriété completeness.*

Preuve. Pour tout couple (Q, G) qui vérifie le prédicat $\Xi(Q, G)$, d'après la construction du protocole GQ2, on a clairement : $\forall (r, d), \langle A_r(Q), B_d(G)\rangle = 1$.

En particulier $Pr_{r,d}(\langle A_r(Q), B_d(G)\rangle) = 1$.

□

2.4.3 GQ2 respecte la propriété «soundness »

La propriété de significativité de seuil de sécurité ϵ se montre en 3 temps :

1. Etablir la probabilité qu'un attaquant arrive à usurper une identité à partir de la technique de l'anticipation du défi avec une probabilité minimale de $1/2^{km}$ (*Etape 1*).
2. Etablir que l'attaquant possède une probabilité non négligeable de générer deux triplets entrelacés valides s'il possède une probabilité supérieure à ϵ de manière non négligeable de s'authentifier (*Etape 2*).
3. Caractériser le seuil de sécurité ϵ (*Etape 3*).

Si on prouve que la probabilité issue de l'anticipation du défi est une probabilité maximale d'usurpation d'identité, c'est à dire $\epsilon = 1/2^{km}$, alors on dit que le protocole GQ2 est **une preuve de connaissance de la factorisation**.

Remarque : Si $\epsilon \neq 1/2^{km}$ alors on ne peut rien déduire si l'attaquant possède une probabilité d'usurpation d'identité comprise entre $1/2^{km}$ et ϵ.

Définition 2.4.1 *Soit n un module composé de deux facteurs premiers p_1, p_2.*
Soient $(g_1, \ldots, g_m)$ les m nombres de base du protocole GQ2.
On définit l'ensemble $Y1$ par :

$$Y1 = \{(e_1, \ldots, e_m) \in \{0, \ldots, 2^k - 1\}^m / h_{p_1}\left(\prod_{i=1}^{m} g_i^{e_i}\right) = h_{p_2}\left(\prod_{i=1}^{m} g_i^{e_i}\right)\}$$

Remarque : cet ensemble caractérisera les $m - uplets$ codés sur k bits qui ne permettent pas de dévoiler la factorisation du module à partir de triplets entrelacés.

Proposition 2.4.2 *Soit $Y1$ l'ensemble défini par la Définition 2.4.1.*
Le protocole GQ2 est un protocole interactif à 3 passes qui vérifie la propriété soundness de seuil de sécurité $\epsilon = \dfrac{|Y1|}{|Y|}$.

Etape 1 : Anticipation du défi

Tout imposteur peut « parier » sur le défi qui sera envoyé par le vérifieur : il a 1 chance sur l'ensemble des défis possibles de s'authentifier, c'est à dire 1 chance sur 2^{km}. En effet, le triplet $\{W^* = D^v \prod_{i=1}^{m} G_i^{d_i} \bmod n, d, D\}$ peut être validé.

La probabilité d'usurper l'identité du prouveur est donc au minimum de $1/2^{km}$.

Etape 2 : Génération de triplets entrelacés valides

Initié par [GMR88], le Forking Lemma (cf Lemme 1.4.1) démontré par D. Pointcheval et J. Stern [PS96] permet dans notre contexte, de déduire une probabilité non négligeable d'extraction de triplets entrelacés valides sous l'hypothèse d'une probabilité ϵ' de réussir l'authentification.

Ainsi, on démontre par l'absurde que si un attaquant arrive à s'authentifier avec une probabilité supérieure à la probabilité ϵ de manière non négligeable, alors il peut construire une machine de Turing probabiliste à temps polynomial qui permet d'extraire la factorisation du module. Ce n'est donc pas un attaquant.

Lemme 2.4.2 *Soit $Y1$ l'ensemble défini par la Définition 2.4.1.*

$$\forall \epsilon' \in [0,1],\ Pr[\langle \tilde{A}_{w_{\tilde{a}}}(G), B_d(G)\rangle] \geq \epsilon' \Rightarrow$$
$$Pr[\langle \tilde{A}_{w_{\tilde{a}}}(G), B_d(G)\rangle, \langle \tilde{A}_{w_{\tilde{a}}}(G), B_{d^*}(G)\rangle, d-d^* \notin Y1] \geq \frac{1}{8}\left(\epsilon' + \frac{|Y1|}{|Y|}\right)\left(\epsilon' - \frac{|Y1|}{|Y|}\right)^2$$

Preuve. On applique le Forking Lemma avec la relation $\xi(w_{\tilde{a}}, d) = \langle \tilde{A}_{w_{\tilde{a}}}(G), B_d(G)\rangle$.
Si $Pr_{w_{\tilde{a}},d}[\langle \tilde{A}_{w_{\tilde{a}}}(G), B_d(G)\rangle] \geq \epsilon'$ alors pour toute valeur $\theta \in [0,1]$,

$$Pr_{w_{\tilde{a}},d,d^*}[\langle \tilde{A}_{w_{\tilde{a}}}(G), B_d(G)\rangle, \langle \tilde{A}_{w_{\tilde{a}}}(G), B_{d^*}(G)\rangle, d-d^* \notin Y1] \geq \left(\theta - \frac{|Y1|}{|Y|}\right)\theta(\epsilon' - \theta)$$

On obtient le résultat pour $\theta = \frac{1}{2}\left(\epsilon' + \frac{|Y1|}{|Y|}\right)$

□

Remarque : La condition $\epsilon' > \frac{|Y1|}{|Y|}$ est une condition suffisante pour minorer la probabilité de générer des triplets entrelacés valides par une fonction non négligeable.

Lemme 2.4.3 *Soit $Y1$ l'ensemble défini par la Définition 2.4.1. Soit ϵ le seuil de sécurité tel que $\epsilon = \frac{|Y1|}{|Y|}$. On a :*

$$\forall c > 0, \exists M \in MT / \forall \tilde{A} \in MT, \exists n_0, \forall |G| > n_0,$$

$$Pr[\langle \tilde{A}_{w_a}(G), B_{w_b}(G)\rangle] \geq \epsilon + \frac{1}{|G|^c} \Rightarrow Pr[\Xi(M(G), G)] \geq 1 - e^{-|G|}$$

Preuve. On remarque que le deuxième membre de l'inégalité du Lemme 2.4.2 est minorée par $\frac{1}{8}\frac{1}{|G|^{3c}}$, en considérant $\epsilon' = \epsilon + \frac{1}{|G|^c}$.

Construisons la machine de Turing M paramétrée par le vérifieur honnête de $GQ2$ noté $B_d(G)$ et par le prouveur malhonnête $\tilde{A}_{w_{\tilde{a}}}(G)$.

Entrée : $G = (n = pq, b = 1, g_1, \ldots, g_m)$
Pour i **de** 1 **à** $N = 8|G|^{3c+1}$ **faire :**
$w_a \leftarrow () ; d \leftarrow () ; d^* \leftarrow ()$
Si $\langle \tilde{A}_{w_{\tilde{a}}}(G), B_d(G) \rangle$ **et** $\langle \tilde{A}_{w_{\tilde{a}}}(G), B_{d^*}(G) \rangle$ et $d^* - d \notin Y1$ **alors**
$(W, d, D) \leftarrow VUE\langle \tilde{A}_{w_{\tilde{a}}}(G), B_d(G) \rangle$
$(W, d^*, D^*) \leftarrow VUE\langle \tilde{A}_{w_{\tilde{a}}}(G), B_{d^*}(G) \rangle$
$\chi^* = \left(\frac{D}{D^*}\right)^{2^k} \prod_{i=1}^{m} g_i^{(d_i - d_i^*)} \bmod n$
$j_0 \leftarrow min\{j : 1, \ldots, b \,/ (\chi^*)^{2^j} = 1 \bmod n\}; \omega \leftarrow (\chi^*)^{2^{j_0 - 1}}$
$p, q \leftarrow pgcd(\omega - 1, n), pgcd(\omega + 1, n)$
Sortir : $\{p, q\}$
Fin si
Fin Pour
Sortir : $NULL$

La machine de Turing $M(G)$ retourne la factorisation du module si l'événement $[\langle \tilde{A}_{w_{\tilde{a}}}(G), B_d(G) \rangle, \langle \tilde{A}_{w_{\tilde{a}}}(G), B_{d^*}(G) \rangle, d - d^* \notin Y1]$ se produit pendant les $8|G|^{3c+1}$ itérations.

La probabilité que cet événement se produise est minorée par la quantité $1 - \left(1 - \frac{1}{8}\frac{1}{|G|^{3c}}\right)^{8|G|^{3c+1}}$, c'est-à-dire par $1 - e^{-|G|}$.

□

Etape 3 : Caractérisation du seuil de sécurité

Selon le nombre de nombres de base, on donne une première estimation (par excès) du seuil de sécurité : $\epsilon = 1/2^k$ si $m = 1$, et $\epsilon = 1 - 2^{-m}$ si $m > 1$.

Proposition 2.4.3 *Soit g le nombre de base unique du protocole GQ2.*
Soient deux triplets entrelacés $\{W, d, D\}$ et $\{W, d^*, D^*\}$.
Alors $\{W, d, D\}$ et $\{W, d^*, D^*\}$ sont des triplets entrelacés valides.

Preuve. Soient deux triplets entrelacés $\{W, d, D\}$ et $\{W, d^*, D^*\}$. On a les égalités :

$$\begin{aligned} W &= D^{2^{k+1}} G^d \bmod n \\ W &= D^{*2^{k+1}} G^{d^*} \bmod n \end{aligned}$$

On peut supposer $d > d^*$ sans perte de généralité.
Soit $l = max\{j / d = d^* \bmod 2^j\}$.
On calcule $(d - d^*)/2^l$ et on applique le théorème de Bezout :
$((d - d^*)/2^l)i + j2 = 1$ avec $(i, j) \in \mathbb{Z}^2$.

$$\begin{aligned} G^{d-d^*} &= (D^*/D)^{2^{k+1}} \bmod n \\ G^{1-j2} &= (D^*/D)^{i2^{k+1-l}} \bmod n \\ G &= (D^*/D)^{i2^{k+1-l}} . G^{j2} \bmod n \\ Q^{2^k} &= (D^*/D)^{i2^{k-l}} . G^j \bmod n \end{aligned}$$

On extrait la factorisation du module n car $pgcd(Q^{2^k} g - 1, n) = p$ *ou* q, d'après le choix fait sur le nombre de base g (car $(g|n) = -1$ et donc $Q^{2^k} \in V_n$ et $g \in (V_p \cap \bar{V}_q) \cup (\bar{V}_p \cap V_q)$).

□

Remarque : Les conditions de sécurité optimale sont respectées dans le cas de l'utilisation d'un nombre de base unique.

Proposition 2.4.4 *Soit $(g_1, \ldots, g_m)$ les m nombres de base du protocole GQ2.*
Soient deux triplets entrelacés $\{W, d, D\}$ et $\{W, d^*, D^*\}$.
Alors $\{W, d, D\}$ et $\{W, d^*, D^*\}$ sont des triplets entrelacés valides avec une probabilité minimale de 2^{-m}, lorsque les variables d et d^* sont indépendantes et suivent une loi uniforme sur l'ensemble $\{0, \ldots, 2^k - 1\}^m$.

Preuve. Soient deux triplets entrelacés $\{W, d, D\}$ et $\{W, d^*, D^*\}$. On a les égalités :

$$W = D^{2^{k+1}} \prod_{i=1}^{m} G_i^{d_i} \bmod n \text{ et } W = D^{*2^{k+1}} \prod_{i=1}^{m} G_i^{d_i^*} \bmod n$$

On suppose $d \neq d^*$ ($d = d_1||..||d_m$ et $d^* = d_1^*||..||d_m^*$).
Soit $l_i = max\{j / d_i = d_i^* \bmod 2^j\}$. On note $d_i - d_i^* = x_i.2^{l_i+1} + 2^{l_i}$, où $x_i = \lfloor (d_i - d_i^*)/2^{l_i+1} \rfloor$, en supposant, sans perte de généralité que $d_i \geq d_i^*$. En effet, si $d_i^* \geq d_i$, il suffit de poser $d_i^* - d_i = -(d_i - d_i^*)$.

Soit $l = min\{l_i / 1 \leq i \leq m\}$

On a alors :

$$\begin{aligned} (D^*/D)^{2^{k+1}} &= \prod_{i=1}^{m} G_i^{d_i - d_i^*} \bmod n \\ (D^*/D)^{2^{k+1}} &= \prod_{i=1}^{m} G_i^{x_i.2^{l_i+1}+2^{l_i}} \bmod n \end{aligned}$$

On élève le tout à la puissance 2^{k-l} :

$$\begin{aligned} (D^*/D)^{2^{2k+1-l}} &= (\prod_{i=1}^{m} G_i^{x_i.2^{l_i+1}+2^{l_i}})^{2^{k-l}} \bmod n \\ (D^*/D)^{2^{2k+1-l}} &= (\prod_{i=1}^{m} G_i^{x_i.2^{l_i+1-l}+2^{l_i-l}})^{2^k} \bmod n \end{aligned}$$

Or $x_i.2^{l_i+1-l} = (d_i - d_i^*)/2^l$ donc

$$(D^*/D)^{2^{2k+1-l}} = (\prod_{i=1}^{m} G_i^{(d_i-d_i^*)/2^l+2^{l_i-l}})^{2^k} \bmod n$$

On isole les G_i où $l_i = l$:

$$\begin{aligned} (D^*/D)^{2^{2k+1-l}} &= (\prod_{i/i=l} G_i \prod_{i/i=l} G_i^{(d_i-d_i^*)/2^l} \prod_{i/i\neq l} G_i^{(d_i-d_i^*)/2^l+2^{l_i-l}})^{2^k} \bmod n \\ (D^*/D)^{2^{2k+1-l}} &= (\prod_{i/i=l} G_i^{2^k}).(\prod_{i} G_i^{(d_i-d_i^*)/2^l}. \prod_{i/i\neq l} G_i^{2^{l_i-l}})^{2^k} \bmod n \end{aligned}$$

On isole $\prod_{i/i=l} G_i^{2^k}$:

$$\prod_{i/i=l} G_i^{2^k} = (D^*/D)^{2^{2k+1-l}}/(\prod_i G_i^{(d_i-d_i^*)/2^l}.\prod_{i/i\neq l} G_i^{2^{l_i-l}})^{2^k} \bmod n$$

$$\prod_{i/i=l} G_i^{2^k} = ((D^*/D)^{2^{k+1-l}}/(\prod_i G_i^{(d_i-d_i^*)/2^l}.\prod_{i/i\neq l} G_i^{2^{l_i-l}}))^{2^k} \bmod n$$

$$\prod_{i/i=l} Q_i^{2^k} = (D^*/D)^{2^{k-l}}/(\prod_i g_i^{(d_i-d_i^*)/2^l}.\prod_{i/i\neq l} g_i^{2^{l_i-l}}) \bmod n$$

On observe $k-l>0$, $l_i-l>0$ car $l_i \neq l$, et $g_i^{(d_i-d_i^*)/2^l} \bmod n$ est calculable.

L'imposteur révèle la factorisation du module n si

$$pgcd(\prod_{i/i=l} Q_i^{2^k} g_i - 1, n) = p \ ou \ q \tag{2.1}$$

D'après le choix fait sur le nombre de base g, une condition suffisante pour que l'imposteur génère des triplets entrelacés valides par le calcul de $pgcd(\prod_{i/i=l} Q_i^{2^k} g_i - 1, n)$ (cf Equation 2.1), est que l'indice du nombre g soit le seul égal à l (car $(g_l|n) = -1$ et donc $Q_l^{2^k} \in V_n$ et $g_l \in (V_p \cap \bar{V}_q) \cup (\bar{V}_p \cap V_q)$).

Montrons que la probabilité de remplir cette condition suffisante est minorée par 2^{-m} : la condition $\epsilon' > \frac{|Y1|}{|Y|} \approx 1 - 2^{-m}$ permet de conclure, grâce au Lemme 2.4.2, à une probabilité non négligeable de générer des triplets entrelacés valides.

Par définition, on a $l = min\{max\{j/d_i = d_i^* \bmod 2^j\}/1 \leq i \leq m\}$.

Le nombre de cas de défis $e_i = d_i - d_i^*$ qui permettent d'extraire la factorisation du module, en parcourant les différentes valeurs de l possibles est la suite géométrique suivante : $\sum_{l=0}^{k-1} (2^m)^l$.

Ainsi, la probabilité pour un attaquant d'obtenir des triplets entrelacés valides est minorée par :

$$\frac{2^{km}-1}{2^m-1}/2^{km} \approx \frac{1}{2^m}$$

□

Le protocole GQ2 est une preuve de connaissance de la factorisation si $\epsilon = 1 - \frac{1}{2^m} = \frac{1}{2^{km}}$ c'est-à-dire si $k = m = 1$.

La sécurité du protocole GQ2 se positionne par rapport au problème difficile de la factorisation de manière d'autant plus étroitement que le seuil de sécurité se rapproche de la probabilité minimale d'usurpation d'identité de $1/2^{km}$.

Dans le chapitre suivant, nous améliorerons l'estimation de ce seuil de sécurité. Dans le cas $m > 1$, nous montrerons en fait que $\epsilon = 1/2$, et $\epsilon < 1/2$ dans le cadre de la généralisation du protocole GQ2 à des facteurs quelconques (cf Partie 3.4.2).

2.4.4 GQ2 respecte la propriété «zero-knowledge »

Proposition 2.4.5 *Le protocole GQ2 est un protocole interactif à 3 passes qui vérifie la propriété zero-knowledge, si* 2^{km} *croît comme un polynôme en* $|n|$.

Il s'agit de montrer que l'observation des échanges entre un vérifieur malhonnête et un prouveur A connaissant la factorisation du module, apporte la même information que les échanges simulés par $\tilde{B}$ à l'aide d'une machines de Turing $M(G)$ ne connaissant pas la factorisation du module.
Mathématiquement, il faut donc construire une machine de Turing $M(G)$ dont la loi de probabilité est indistinguable de la variable aléatoire $VUE\langle A_r(Q), \tilde{B}_{w_d}(G)\rangle$ (cf Définition 1.4.3) :

Lemme 2.4.4 *Les variables aléatoires* $VUE\langle A_r(Q), \tilde{B}_d(G)\rangle$ *et* $M_{d,d^*,D}(G)$ *sont indistinguables.*

Preuve. Le Tableau 2.4 indique le mode de génération des deux valeurs aléatoires dont nous montrons qu'elles sont indistinguables : un observateur ne peut distinguer des triplets valides issus d'un mode de génération privé, d'un mode de génération public.

$VUE\langle A_r(Q), \tilde{B}_d(G)\rangle$:	$M_{d,d^*,D}(G)$:
	Répéter 2^{km} fois
$r \leftarrow \mathbb{Z}_n()$	$d^* = (d_1^*, \ldots, d_m^*) \leftarrow \{0, \ldots, 2^k - 1\}^m()$
$W \leftarrow r^{2^{k+1}} \bmod n$	
$d = (d_1, \ldots, d_m) \leftarrow \{0, \ldots, 2^k - 1\}^m()$	$D \leftarrow \mathbb{Z}_n()$
$d^* = (d_1^*, \ldots, d_m^*) \leftarrow E(d, W)$	$d = (d_1, \ldots, d_m) \leftarrow \{0, \ldots, 2^k - 1\}^m()$
$D \leftarrow r \times \prod_{i=1}^{m} Q_i^{d_i^*} \bmod n$	$W \leftarrow D^{2^{k+1}} \times \prod_{i=1}^{m} g_i^{2d_i^*} \bmod n$
	Si $d^* = E(d, W)$ **alors** Retourner (W, d^*, D) Fin si
Retourner (W, d^*, D)	**Fin répéter**
	Retourner NULL
r, D et d^* sont des variables aléatoires uniformes, d une variable aléatoire de loi définie par $\tilde{B}$ et E une fonction définie par $\tilde{B}$. Toutes les variables aléatoires sont supposées indépendantes. De plus, la dépendance de d et E par rapport à $\tilde{B}$ représente le caractère malhonnête de $\tilde{B}$	

TAB. 2.4 – Deux variables aléatoires indistinguables

Distribution de $VUE\langle A_r(Q), \tilde{B}_d(G)\rangle$:

$$Pr_{r,d}[VUE\langle A_r(Q), \tilde{B}_d(G)\rangle = (x, y = (y_1, \ldots, y_m), z)] =$$
$$Pr_{r,d}[r^{2^{k+1}} \bmod n = x, E(d, r^{2^{k+1}} \bmod n) = y, r \times \prod_{i=1}^{m} Q_i^y \bmod n = z]$$
$$Pr_{r,d}[z^{2^{k+1}} \times \prod_{i=1}^{m} g_i^{2^b y_i} \bmod n = x, E(d, x) = y, r = z \times \prod_{i=1}^{m} Q_i^{-y} \bmod n]$$

Donc,

$$Pr_{r,d}[VUE\langle A_r(Q), \tilde{B}_d(G)\rangle = (x, y = (y_1, \ldots, y_m), z)\rangle = \begin{cases} 0 \text{ si } z^{2^{k+1}} \prod_{i=1}^{m} g_i^{2^b y_i} \bmod n \neq x \\ \frac{Pr_d[E(d,x)=y]}{n} \text{ sinon} \end{cases}$$

Distribution de $M_{d,d^*,D}(G)$:

L'événement $[M_{d,d^*,D}(G) = (x, y = (y_1, \ldots y_m), z)]$ se réalise si et seulement si l'événement suivant se réalise au moins une fois au cours des 2^{km} itérations :

$$A = [z^{2^{k+1}} \times \prod_{i=1}^{m} g_i^{2^b y_i} \bmod n = x, d^* = y, E(d, x) = y, D = z]$$

Si $z^{2^{k+1}} \prod_{i=1}^{m} g_i^{2^b y_i} \bmod n \neq x$ alors $P[A] = 0$
Sinon,

$$P[A] = Pr_d[E(d, x) = y] Pr_{d^*}[d^* = y] Pr_D[D = z] = \frac{1}{n} \frac{1}{2^{km}} Pr_d[E(d, x) = y]$$

$$Pr_{r,d}(M_{d,d^*,D}(G) = (x, y = (y_1, \ldots y_m), z)) = \begin{cases} 0 \text{ si } z^{2^{k+1}} \prod_{i=1}^{m} g_i^{2^b y_i} \bmod n \neq x \\ 1 - \left(1 - \frac{Pr_d[E(d,x)=y]}{2^{km} n}\right)^{2^{km}} \text{ sinon} \end{cases}$$

La relation $\forall x \in [0, 1], \forall n \in \mathbb{N}, 1 - x \leq (1 - \frac{x}{n})^n \leq 1 - x + \frac{x^2}{2}$ fournit la probabilité plus générale suivante :

$$\sum_{(x,y,z)} |Pr_{r,d}[VUE\langle A_r(Q), \tilde{B}_d(G)\rangle = (x, y, z)] - Pr_{d,d^*,D}[(M_{d,d^*,D}(G)) = (x, y, z)]|$$
$$\leq \frac{1}{2} \sum_{(x,y)} \left(\frac{1}{n} Pr_d[E(d, x) = y]\right)^2 \leq \frac{1}{2n}$$

La propriété zero-knowledge résulte de la propriété que la quantité $\frac{1}{2n}$ décroît exponentiellement en $|n|$.

□

Chapitre 3

Généralisation du protocole GQ2

Début 2004, un nouveau brevet déposé par L. Guillou et JJ. Quisquater, généralise le protocole d'authentification et de signature GQ2 à tout module. Ce chapitre est issu d'un travail de collaboration avec Francois Daudé et Louis Guillou.

L'idée d'adapter le protocole à tout module est née de la constatation suivante : GQ2 peut être considéré comme une brique de sécurité complémentaire à RSA. En effet, GQ2 est une solution de sécurité plus intéressante dans certains contextes d'intégration à contraintes fortes, de coût et de performance ; tout en maintenant le lien avec le problème difficile de la factorisation des grands nombres. La mise en place d'un procédé de génération de clés privées GQ2 à partir de n'importe quelle clé RSA existante, est alors décrite : il existe une probabilité écrasante de construire des clés GQ2 à partir d'un grand nombre entier quelconque, composé de facteurs premiers quelconques.

L'objectif de cette généralisation du protocole GQ2 est le maintien de l'infrastructure utilisée par RSA dans les systèmes existants afin de permettre l'intégration du protocole GQ2, et de bénéficier de ses performances, quels que soient les modules. Or, une application directe du protocole avec des modules quelconques, tel que décrit dans le chapitre précédent, entraîne une réduction importante de la probabilité de trouver le petit nombre public de la clé publique qui permet de vérifier l'équivalence entre le prédicat associé à GQ2 et celui de la factorisation. Ainsi, une généralisation de l'équation générique est donc proposée pour permettre une recherche avec une probabilité écrasante de ce nombre, qui atteste de la preuve de sécurité d'équivalence du schéma avec la factorisation des grands nombres.

La deuxième partie de ce chapitre se tourne vers la possibilité d'utiliser des modules multifacteurs et de généraliser l'exposant public. Les résultats de comparaison des performances du protocole, en fonction des variations de ces nouveaux paramètres, donnent des résultats satisfaisants face au protocole référence de Fiat-Shamir.

Dans la suite, nous regarderons une autre application des techniques mises en place pour la généralisation du protocole GQ2 : dans le domaine de la signature électronique, le protocole Rabin-Williams est également une preuve de la connaissance de la factorisation du module.

3.1 Spécifications du protocole

On généralise les spécifications introduites dans la Partie 2.2 qui présentent les étapes de l'intégration du protocole GQ2.

3.1.1 Elaboration des bi-clés à partir de deux facteurs quelconques

De la même façon que pour le protocole initial, les paramètres k et m sont les paramètres qui déterminent le niveau de sécurité d'un tour d'exécution. Un nouveau paramètre dit **d'adaptation** b, vient s'ajouter aux paramètres de la clé publique. L'exposant de vérification est alors défini par $v = 2^{k+b}$, où b est le niveau de n.

La construction d'une bi-clé GQ2 généralisée s'effectue alors selon les étapes suivantes :

- On choisit aléatoirement deux nombres p_1 et p_2 **premiers quelconques**
- On calcule le module n égal au produit de p_1 par p_2
- La clé publique GQ2 se compose du module n et de m nombres publics notés $(G_1, ..., G_m)$, chaque G_i étant la puissance $2^{b\ ieme}$ d'un petit nombre premier noté $g_i : \forall i \in \{1, ..., m\},\ G_i = g_i^{2^b}$
- La clé publique GQ2 doit vérifier la propriété suivante :
  Pour au moins un nombre de base, noté g, nous avons $(g|p_1) = -(g|p_2)$ si $b_{p_1} = b_{p_2}$, et $(g|p_1) \neq 1$ si $b_{p_1} > b_{p_2}$ sans perte de généralité, où b_i est le niveau de p_i
- La clé privée GQ2 se compose des nombres premiers p_1, p_2 et de m nombres secrets notés $(Q_1, ..., Q_m)$ reliés aux nombres publics par les équations génériques suivantes :

$$\forall i \in \{1, ..., m\}, G_i Q_i^v = 1 \bmod n$$

3.1.2 Le protocole d'authentification

Le protocole GQ2 s'effectue entre un prouveur et un vérifieur. Le vérifieur connaît la clé publique $(n, G_1, ..., G_m)$ et le prouveur connaît la clé privée $(p_1, p_2, Q_1, ..., Q_m)$. Ils possèdent en commun l'exposant de vérification $v = 2^{k+b}$ et le paramètre de multiplicité m.

Le prouveur GQ2 réalise alors systématiquement les étapes suivantes :

1. Sélection d'un nombre aléatoire positif et inférieur à n, noté r
2. Calcul de $W = r^v \bmod n$
3. En réponse à un défi émis par le vérifieur, consistant en m nombres aléatoires de k-bits notés $(d_1, ..., d_m)$, calcul du nombre $D = r.Q_1^{d_1}...Q_m^{d_m} \bmod n$
4. Effacement du nombre aléatoire r.

Le vérifieur réalise systématiquement les étapes suivantes :

1. Réception de l'engagement W
2. Sélection de m nombres aléatoires de k-bits notés $(d_1, ..., d_m)$
3. En réponse à un nombre D émis par le prouveur, calcul du nombre $W' = D^v.G_1^{d_1}...G_m^{d_m} \bmod n$ et vérification de la condition $W' = W$
4. Si la condition précédente est vérifiée, acceptation du prouveur.

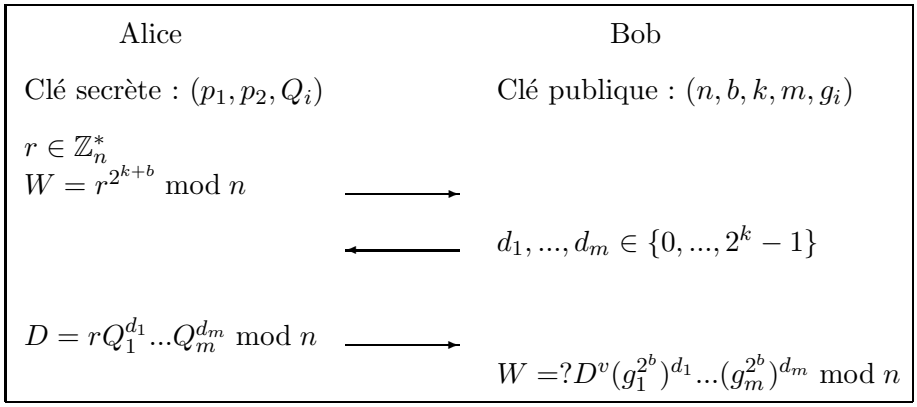


FIG. 3.1 – Schéma du protocole GQ2 pour $v = 2^{k+b}$

3.1.3 Le protocole de signature

Comme tout schéma de signature provenant d'un schéma d'authentification à 3 passes à divulgation nulle de connaissance, le protocole de signature de GQ2 se déduit du schéma d'authentification. Soit h une fonction de hachage et M un message à signer.

1. Génération des clés :
   Soient p_1 et p_2 deux nombres premiers et $n = p_1 p_2$.
   Clé publique pk : $(n, G_1, ..., G_m)$.
   Clé privée sk : $(p_1, p_2, Q_1, ..., Q_m)$.
   L'exposant de vérification $v = 2^{k+b}$ et le paramètre de multiplicité m sont des données partagées par le signataire et le vérifieur.
2. Génération de la signature : $Sign(M, pk, sk)$
   - Sélection d'un nombre aléatoire positif et inférieur à n, noté r
   - Calcul de $W = r^v \bmod n$
   - Calcul de $h(W, M)$ et extraction de m nombres de k-bits notés $(d_1, ..., d_m)$
   - Calcul de la signature $s = r.Q_1^{d_1}...Q_m^{d_m} \bmod n$ et envoi de $(d_1, \ldots, d_m, s)$
   - Effacement du nombre aléatoire r.

3. Vérification de la signature : $Verif(s, M, pk)$
   Le vérifieur réalise systématiquement les étapes suivantes :
   (a) Réception de $(d_1, \ldots, d_m, s)$
   (b) Calcul du nombre $W' = s^v.G_1^{d_1}...G_m^{d_m} \bmod n$
   (c) Calcul de $(d'_1, ..., d'_m)$ à partir du calcul de $h(W', M)$
   (d) Si $(d_1, ..., d_m) = (d'_1, ..., d'_m)$, validation de la signature, et rejet sinon.

L'expérience de l'intégration des deux protocoles de signature RSA et GQ2 dans une carte bas coût avec une taille de clés de 1024 bits, rend compte du rapport de performance RSA/GQ2 : **10** ($k = 16$, $m = 10$).

3.2 Gain pour la recherche des nombres de base

Dans cette partie, nous présentons le calcul des clés privées quelle que soit la nature des facteurs qui composent le module. En utilisant la Proposition 1.2.6, nous généralisons le calcul des racines carrées d'ordre impair introduit pour le cas particulier des facteurs congrus à 3 modulo 4 dans le Chapitre 2 (cf Proposition 2.2.1).

La généralisation de l'équation générique sera justifiée par l'amélioration des probabilités de réussite pour la recherche du nombre de base dans le cas de 2 facteurs quelconques. Dans le chapitre précédent, le choix de modules composés de facteurs congrus à 3 modulo 4 impliquait la probabilité d'1 chance sur 2 d'obtenir un nombre de base permettant d'obtenir l'équivalence entre le prédicat associé à GQ2 et celui associé à la factorisation. Nous obtenons ici la même probabilité grâce à l'ajout du paramètre d'adaptation dans le cas de facteurs quelconques.

3.2.1 Génération des clés privées

La génération des clés privées peut se faire à partir des calculs respectifs dans chacun des groupes multiplicatifs des facteurs, ou bien directement dans le groupe multiplicatif du module.

Proposition 3.2.1 *Soient $p_1, p_2 \in \mathbb{P}$ tels que $n = p_1 p_2$. Chaque facteur p_i s'écrit sous la forme $p_i = 2^{b_i} q_i + 1$ avec q_i impair, et on a $b = max\{b_1, b_2\}$.*

La valeur privée Q_i de l'équation générique $g^{2^b} Q_i^{2^{k+b}} = 1 \bmod p_i$ se calcule par $Q_i = g^{-2^b . x_i} \bmod p_i$, où

$$x_i = ((p_i + 2^{b_i} - 1)/2^{b_i+1})^{k+b} \bmod (p_i - 1)/2^{b_i}$$

Preuve. Il suffit de vérifier l'équation générique $GQ_i^v = 1 \bmod p_i$:

$$Q_i^v = (g^{-2^b . x_i})^{2^{k+b}} = g^{-2^b((p_i+2^{b_i}-1)/2^{b_i+1})^{k+b}.2^{k+b}} = g^{-2^b((q_i+1)/2)^{k+b}.2^{k+b}} \bmod p_i$$

$$= g^{-2^b(q_i+1)^{k+b}/2^{k+b}.2^{k+b}} = g^{-2^b(q_i+1)^{k+b}} = (...((g^{-2^b})^{q_1+1})^{q_1+1}...)^{q_1+1} \bmod p_i$$

D'après la Proposition 1.2.5, si a est d'ordre impair dans $\mathbb{Z}_{p_i}^*$, alors $a^{q_i+1} = a \bmod p_i$. Comme g^{2^b} est d'ordre impair, alors g^{-2^b} également, et donc, on a bien $Q_i^v = g^{-2^b} \bmod p_i$.

□

Proposition 3.2.2 *Soient $p_1, p_2 \in \mathbb{P}$ tels que $n = p_1 p_2$. Chaque facteur p_i s'écrit sous la forme $p_i = 2^{b_i} q_i + 1$ avec q_i impair, et on a $b = max\{b_1, b_2\}$.*

La valeur privée Q de l'équation générique $g^{2^b} Q^{2^{k+b}} = 1 \bmod n$ se calcule par $Q = g^{-2^b . y} \bmod n$, où

$$y = (q_1 q_2 + 1)/2^{k+b} \bmod (p_1 - 1)(p_2 - 1)/2^{(b_1+b_2)}$$

Preuve. Comme la Proposition 3.2.1, il suffit de vérifier l'équation générique $GQ^v = 1 \bmod n$:

$$Q_i^v = (g^{-2^b . y})^{2^{k+b}} = g^{-2^b . (q_1 q_2+1)/2^{k+b}.2^{k+b}} = g^{-2^b . (q_1 q_2+1)} = (g^{-2^b})^{q_1 q_2+1}$$

$$= (...((g^{-2^b})^{q_1q_2+1})^{q_1q_2+1}...)^{q_1q_2+1} \bmod n$$

De même que précédemment, si a est d'ordre impair dans $\mathbb{Z}_n^*$, alors $a^{q_1q_2+1} = a \bmod n$. Comme g^{2^b} est d'ordre impair, alors g^{-2^b} également, et donc, on a bien $Q_i^v = g^{-2^b} \bmod n$.

□

3.2.2 Passage de la version restrictive à la version généralisée

Nous nous concentrons sur la recherche du nombre de base g qui assure l'équivalence du prédicat associé à la connaissance des clés privées GQ2 avec celui de la factorisation du module.

Cette recherche est basée sur l'équation générique. Sans l'ajout du paramètre d'adaptation b, on a la proposition suivante :

Proposition 3.2.3 *Soient p_1, p_2 deux facteurs tels que $n = p_1p_2$, et V_{p_i} l'ensemble défini par la Proposition 1.2.2.*
Soient $g, Q \in \mathbb{Z}_n^*$ tels que $g^2Q^{2^{k+1}} = 1 \bmod n$.
On a l'équivalence entre les deux prédicats de la Définition 1.4.2 si $g^2 \in V_n$ et $g \in (V_{p_1} \cap \bar{V}_{p_2}) \cup (\bar{V}_{p_1} \cap V_{p_2})$.

L'équation générique intégrant le paramètre b induit une nouvelle proposition pour obtenir des conditions suffisantes de recherche du nombre de base g :

Proposition 3.2.4 *Soient p_1, p_2 deux facteurs tels que $n = p_1p_2$, et V_{p_i} l'ensemble défini par la Proposition 1.2.2.*
Soient $g, Q \in \mathbb{Z}_n^*$ tels que $g^{2^b}Q^{2^{k+b}} = 1 \bmod n$ où b est le niveau de n.
On a l'équivalence entre les deux prédicats de la Définition 1.4.2 s'il existe $0 < \alpha \leq b$ tel que $g^{2^\alpha} \in V_n$ et $g^{2^{\alpha-1}} \in (V_{p_1} \cap \bar{V}_{p_2}) \cup (\bar{V}_{p_1} \cap V_{p_2})$.

Remarque : Dans le cas de facteurs congrus à 3 modulo 4, les deux propositions précédentes sont équivalentes.

Dans le cadre de la norme ISO/IEC 9798-5 [ISO04], les conditions de recherche sont moins restrictives afin d'obtenir des conditions uniquement sur les symboles de Legendre :

Corollaire 3.2.1 *Soient p_1, p_2 deux facteurs tels que $n = p_1p_2$.*
Soient $g, Q \in \mathbb{Z}_n^*$ tels que $g^{2^b}Q^{2^{k+b}} = 1 \bmod n$ où b est le niveau de n.
On a l'équivalence entre les deux prédicats de la Définition 1.4.2 si :
– *$(g|p_1) = -(g|p_2)$ dans le cas $b_{p_1} = b_{p_2}$ (en particulier dans le cas $b_{p_1} = b_{p_2} = 1$)*
– *$(g|p_1) \neq 1$ dans le cas $b_{p_1} > b_{p_2}$.*

Dans le cas de facteurs congrus à 3 modulo 4, on a l'équivalence $a \in V_{p_i} \Leftrightarrow (g|p_i) = 1$, donc, d'après la Proposition 3.2.3 ou 3.2.4 et la proportion de résidus quadratiques dans un corps (cf Théorème 1.2.1), la probabilité de trouver un nombre de base g dans un ensemble à i éléments, est donc de $1 - 2^{-i}$.

Dans le cas de facteurs quelconques, la condition suffisante du Corollaire 3.2.1 permet également d'en déduire le même résultat de probabilité.

Dans le paragraphe suivant, l'étude des probabilités à partir de la notion de «niveau d'un élément » va être décrite pour permettre d'affiner la recherche. Le paramètre d'adaptation va alors prendre toute son importance.

3.2.3 Une modification minime aux conséquences avantageuses

Dans cette partie, nous calculons la probabilité de trouver un nombre de base qui vérifie la Proposition 3.2.3 ou la Proposition 3.2.4 dans le cas de facteurs quelconques, lorsque l'on considère un petit ensemble de nombres premiers, afin de constater le gain que nous obtenons.

Nous introduisons également le cas de 3 facteurs, et recherchons ainsi deux nombres de base qui permettent d'extraire deux décompositions du module : l'ajout du paramètre b est également favorable à ce cas.

Cas pour deux facteurs

Définition 3.2.1 *On définit l'application notée h par :*

$$\begin{array}{rll} h: & \mathbb{P} & \rightarrow \mathbb{N}^* \\ & p & \mapsto h(p) = max\{i/(p-1)/2^i \in \mathbb{N}^*\} \end{array}$$

Remarque : $h(p) = b_p$ où b_p est le niveau de p.

Proposition 3.2.5 *Soient $p_1, p_2 \in \mathbb{P}$ et b_1, b_2 les niveaux de p_1 et p_2.*
Si on définit $P_{facteurs}$ par

$$P_{facteurs}(b_1, b_2) \quad = \quad Pr(p_1 \leftarrow \mathbb{P}(), p_2 \leftarrow \mathbb{P}(), h(p_1) = b_1 \text{ et } h(p_2) = b_2)$$

alors on a $P_{facteurs}(b_1, b_2) = 1/2^{b_1+b_2}$.

Preuve. Soit $p_i \in \mathbb{P}$. Le calcul de la probabilité que $p_i - 1$ soit divisible par 2^{b_i} et pas par 2^{b_i+1} correspond à rechercher la probabilité de trouver aléatoirement la séquence de $b_i + 1$ bits de poids faibles suivante 10...0 de $p_i - 1$, parmi l'ensemble des séquences de $b_i + 1$ bits, le dernier bit étant toujours égal à 0.

La génération des 2 facteurs se fait de manière indépendante, donc la probabilité de générer deux facteurs aléatoirement de niveau b_1 et b_2 est $1/2^{b_1+b_2}$.

□

Définition 3.2.2 *Soit $p \in \mathbb{P}$. Soit g un élément du groupe multiplicatif $\mathbb{Z}_p^*$ et soit l'application η_p définie par la Proposition 1.2.2.*
*On appelle **niveau de l'élément** g dans $\mathbb{Z}_p^*$, le nombre $h_p(g)$ tel que :*

$$h_p(g) = min\{i \in \mathbb{N}/\eta_p(g)^{2^i} = 1 \bmod p\}$$

Proposition 3.2.6 *Soient $g, g_1, g_2 \in \mathbb{Z}_p^*$ et soit la notion de niveau d'un élément définie par la Définition 3.2.2. On a les propriétés suivantes :*
- *$h_p(g^{2^i}) = max(h_p(g) - i, 0)$ et $h_p(g) = h_p(\eta_p(g))$*

– $h_p(g_1.g_2) = max(h_p(g_1), h_p(g_2))$ *si* $h_p(g_1) \neq h_p(g_2)$ $h_p(g_1 g_2) < h_p(g_1)$ *sinon.*

Preuve. Par définition, $h_p(g)$ désigne le plus petit entier l tel que $\eta(g)^{2^l} = 1 \bmod p$
Supposons sans restriction de généralité que $h_p(g_1) \geq h_p(g_2)$ alors $h_p(g_1 g_2) \leq h_p(g_1)$ puisque $\eta(g_1 g_2)^{2^{h_p(g_1)}} = \eta(g_1)^{2^{h_p(g_1)}} (\eta(g_2)^{2^{h_p(g_2)}})^{2^{h_p(g_1)-h_p(g_2)}} = 1 \bmod p$

– si $h_p(g_1) > h_p(g_2)$ alors
$\eta(g_1 g_2)^{2^{h_p(g_1)-1}} = \eta(g_1)^{2^{h_p(g_1)-1}} (\eta(g_2)^{2^{h_p(g_2)}})^{2^{h_p(g_1)-1-h_p(g_2)}} = \eta(g_1)^{2^{h_p(g_1)-1}} \neq 1 \bmod p$
On a donc $h_p(g_1 g_2) = h_p(g_1)$
– si $h_p(g_1) = h_p(g_2)$ alors
$\eta(g_1 g_2)^{2^{h_p(g_1)-1}} = \eta(g_1)^{2^{h_p(g_1)-1}} (\eta(g_2)^{2^{h_p(g_2)-1}}) = (-1) \times (-1) = 1 \bmod p$
On a donc $h_p(g_1 g_2) < h_p(g_1)$.

□

Lemme 3.2.1 *Soit* $p \in \mathbb{P}$ *et* b_p *le niveau de* p. *Soit* $a \in \mathbb{N}$ *tel que* $0 < a \leq b_p$.
On a $Pr(g \leftarrow \mathbb{Z}_p^*(), h_p(g) = a) = 2^{a-1}/2^{b_p}$ *et* $Pr(g \leftarrow \mathbb{Z}_p^*(), h_p(g) = 0) = 1/2^{b_p}$.

Preuve. D'après la Proposition 3.2.6, $h_p(g) = h_p(\eta_p(g))$ et donc $Pr(g \leftarrow \mathbb{Z}_p^*(), h_p(g)) = Pr(g \leftarrow \mathbb{Z}_p^*(), h_p(\eta_p(g)))$. De plus, le 2-sous groupe de Sylow de $\mathbb{Z}_p^*$ est d'ordre 2^{b_p}, donc

– Si $h_p(g) = 0$ alors $\eta_p(g) = 1$ et donc $Pr(g \leftarrow \mathbb{Z}_p^*(), h_p(g) = 0) = 1/2^{b_p}$
– Sinon $Pr(g \leftarrow \mathbb{Z}_p^*(), h_p(\eta_p(g)) = a) = 2^{a-1}/2^{b_p}$ car il existe 2^{a-1} racines $2^{a\ ieme}$ non triviales de l'unité.

□

Définition 3.2.3 *Soient* $p_1, p_2 \in \mathbb{P}$ *et* b_1, b_2 *les niveaux de* p_1 *et* p_2.
On définit P_{echec} *par la probabilité de ne pas valider les conditions des Propositions 3.2.3 ou 3.2.4 pour un nombre premier* g *choisi au hasard.*

Lemme 3.2.2 *Soient* $p_1, p_2 \in \mathbb{P}$ *et* b_1, b_2 *les niveaux de* p_1 *et* p_2.
Soit P_{echec} *définie par la Définition 3.2.3.*
P_{echec} *pour la Proposition 3.2.3 est :*

$$\begin{aligned} P_{echec}(b_1, b_2) &= Pr(g \leftarrow \mathbb{P} \cup \{2\}(), h(p_1) = b_1, h(p_2) = b_2, \\ &\quad h_{p_1}(g) = h_{p_2}(g) \text{ ou } h_{p_1}(g) > 1 \text{ ou } h_{p_2}(g) > 1) \end{aligned}$$

P_{echec} *pour la Proposition 3.2.4 est :*

$$P_{echec}(b_1, b_2) = Pr(g \leftarrow \mathbb{P} \cup \{2\}(), h(p_1) = b_1, h(p_2) = b_2, h_{p_1}(g) = h_{p_2}(g))$$

Les probabilités des deux conditions des Propositions 3.2.3 et 3.2.4 se calculent de façon identique : on note P_{gen}^i (cf Tableau 3.1) la probabilité de trouver un nombre de base g dans un ensemble à i éléments pour des facteurs choisis au hasard.

$$\begin{aligned} P_{gen}^i &= \sum_{b_1, b_2} P_{cond}^i(b_1, b_2) \\ &= \sum_{b_1, b_2} P_{success}^i(b_1, b_2) \times P_{facteurs}(b_1, b_2) \end{aligned}$$

Dans le cas de 2 facteurs, $P^i_{success}(b_1, b_2) = 1 - (P_{echec}(b_1, b_2))^i$

Le choix de l'ensemble des i candidats considérés n'a pas d'influence sur les résultats : les probabilités sont équiprobables grâce au caractère premier de ces nombres. Cependant, le nombre 2 possède des propriétés particulières qui le conduisent, par exemple, à ne jamais pouvoir être un candidat lorsque les deux facteurs ont un niveau égal à 2 (car 2 est non résidu quadratique dans les 2 cas).

	$f=2$		$f=3$	
m	$v=2^{k+1}$	$v=2^{k+b}$	$v=2^{k+1}$	$v=2^{k+b}$
	$P^i_{gen} \times 100$			
1	22.222	74.074	-	-
2	36.281	90.521	6.9971	34.2252
3	45.732	95.990	13.8801	52.0476
4	52.467	98.174	19.6607	62.2300
5	57.516	99.136	24.4819	68.7385
6	61.462	99.582	28.5830	73.2713
7	64.649	99.795	32.1390	76.6285

TAB. 3.1 – Probabilités de succès pour le cas de 2 et 3 facteurs

Cas pour trois facteurs

Pour 3 facteurs, deux nombres de base sont nécessaires pour appliquer 2 fois le principe universel (cf Proposition 1.3.1), et ainsi extraire tous les facteurs du module.

Proposition 3.2.7 *Soit n un module composé de trois facteurs p_1, p_2, p_3.*

Soient $g_i, Q_i \in \mathbb{Z}_n^*$ tels que $g_i^{2^b} Q_i^{2^{k+b}} = 1 \bmod n$ pour $i = 1, 2$.

On a l'équivalence entre les deux prédicats de la Définition 1.4.2 adaptée à 3 facteurs, s'il existe $0 < \alpha_1 \leq b$ et $0 < \alpha_2 \leq b$ tels que $g_1^{2^{\alpha_1}} \in V_n$, $g_2^{2^{\alpha_2}} \in V_n$ et l'un des 3 cas suivants :

- $g_1^{2^{\alpha_1 - 1}} \in (V_{p_1} \cap \bar{V}_{p_2} \cap \bar{V}_{p_3}) \cup (\bar{V}_{p_1} \cap V_{p_2} \cap V_{p_3})$ *et*
  $g_2^{2^{\alpha_2 - 1}} \in (V_{p_1} \cap \bar{V}_{p_2} \cap V_{p_3}) \cup (\bar{V}_{p_1} \cap V_{p_2} \cap \bar{V}_{p_3})$
- $g_1^{2^{\alpha_1 - 1}} \in (V_{p_1} \cap \bar{V}_{p_2} \cap \bar{V}_{p_3}) \cup (\bar{V}_{p_1} \cap V_{p_2} \cap V_{p_3})$ *et*
  $g_2^{2^{\alpha_2 - 1}} \in (V_{p_1} \cap V_{p_2} \cap \bar{V}_{p_3}) \cup (\bar{V}_{p_1} \cap \bar{V}_{p_2} \cap V_{p_3})$
- $g_1^{2^{\alpha_1 - 1}} \in (V_{p_1} \cap \bar{V}_{p_2} \cap V_{p_3}) \cup (\bar{V}_{p_1} \cap V_{p_2} \cap \bar{V}_{p_3})$ *et*
  $g_2^{2^{\alpha_2 - 1}} \in (V_{p_1} \cap V_{p_2} \cap \bar{V}_{p_3}) \cup (\bar{V}_{p_1} \cap \bar{V}_{p_2} \cap V_{p_3})$

Les colonnes 3 et 4 du Tableau 3.1 répertorient ces résultats.

Dans le cas de 3 facteurs congrus à 3 modulo 4, la probabilité $P^i_{cond}(1, 1, 1)$ d'obtenir deux nombres de base qui vérifient les conditions de la Proposition 3.2.7 dans un ensemble à i éléments est répertoriée dans le Tableau 3.2.

Remarque : on a 12,5% de chance de choisir aléatoirement 3 facteurs congrus à 3 modulo 4. Ces modules représentent la plus grande proportion de modules composés de 3 facteurs.

i	2	3	4	5	6	7	8	9	10
$f = 3$	37.50	65.625	82.031	90.820	95.361	97.668	98.831	99.415	99.707

TAB. 3.2 – Pourcentages de succès pour le cas de 3 facteurs congrus à 3 mod 4

3.2.4 Problématique pour f facteurs

Le Tableau 3.3 montre qu'il est facile de généraliser la probabilité d'échec à f facteurs. Cependant, il n'en est pas de même pour la probabilité de réussite. Ainsi, une approche approximative du problème pour f facteurs va être décrite dans la suite.

f	P_{echec}	
	$v = 2^{k+1}$	$v = 2^{k+b}$
2	$P_{echec}(b_1, b_2) = \frac{2^{b_1+b_2} - 2}{2^{b_1+b_2}}$	$P_{echec}(b_1, b_2) = \frac{2}{2^{b_1+b_2}} + \sum_{i=2}^{min(b_1,b_2)} \frac{(2^{i-1})^2}{2^{b_1+b_2}}$
3	$P_{echec}(b_1, b_2, b_3) = \frac{2^{b_1+b_2+b_3} - 6}{2^{b_1+b_2+b_3}}$	$P_{echec}(b_1, b_2, b_3) = \frac{2}{2^{b_1+b_2+b_3}} + \sum_{i=2}^{min(b_1,b_2,b_3)} \frac{(2^{i-1})^3}{2^{b_1+b_2+b_3}}$
f	$P_{echec}(b_1, b_2, ..., b_f) = \frac{2^{\Sigma b_i} - (2^f - 2)}{2^{\Sigma b_i}}$	$P_{echec}(b_1, b_2, ..., b_f) = \frac{2}{2^{\Sigma b_i}} + \sum_{i=2}^{min(b_1,b_2,...,b_f)} \frac{(2^{i-1})^f}{2^{\Sigma b_i}}$

TAB. 3.3 – Problématique pour f facteurs

La prise en compte de modules de tout niveau a entraîné une adaptation de l'équation générique du protocole GQ2 afin de maintenir les résultats de recherche du nombre de base g, pour un module composé de deux facteurs. De la même façon, l'étude de la recherche de deux nombres de base pour un module composé de 3 facteurs a été faite : parmi l'ensemble des premiers nombres premiers, la probabilité de trouver ces nombres est grande ; en particulier en observant le cas de 3 facteurs congrus à 3 modulo 4. Au-dessus de 3 facteurs, nous simulons cette recherche afin de constater de bons résultats également.

3.3 Extension aux modules multifacteurs

Dans cette partie, nous proposons de définir des notations afin d'établir des techniques d'approximation des probabilités de trouver de « bons » nombres de base pour des modules composés de plusieurs facteurs. Comme le Tableau 2.3 l'indique, l'augmentation du nombre de facteurs qui composent le module, entraîne de meilleures performances pour le temps d'exécution du protocole grâce à la technique des restes chinois. De nouvelles conditions suffisantes sont appliquées afin d'appliquer la recherche des nombres de base quel que soit le nombre de facteurs du module, à travers deux

techniques de simulation. Le choix de petits nombres de base premiers est également justifié.

3.3.1 Notations multifacteurs

Définition 3.3.1 *Soit n un module composé de f facteurs.*
Soient $b_1, b_2...b_f$ les f niveaux respectifs des facteurs.
On pose $k_n = card\{b_1, b_2...b_f\}$.
*On définit une suite finie $h_1, h_2...h_{k_n}$ des **niveaux différents** des facteurs, que l'on ordonne par ordre strictement décroissant : $h_1 > h_2 > ... > h_{k_n}$.*
*On retrouve le **niveau du module** n où $b = max\{b_1, b_2, ..., b_f\} = h_1$.*

Remarque : On a toujours $k_n \leq f$.

Définition 3.3.2 *Soit n un module composé de f facteurs.*
Soit k_n le nombre de niveaux différents des facteurs.
*On appelle **ensemble de niveau du module** n, l'ensemble S_i des facteurs de niveau i.*

$$S_i = \{p \in \mathbb{P}/p|n \text{ et } h(p) = i\}$$

On note $|S_i|$ le nombre d'éléments de l'ensemble S_i.
*En particulier, on appelle **ensemble de niveau minimal** l'ensemble S_{k_n}.*

Définition 3.3.3 *Soit n un module composé de f facteurs.*
Soit k_n le nombre de niveaux différents des facteurs.
*On définit les **sous-modules** $n_1, n_2...n_{k_n}$ du module n, les entiers positifs tels que :*

$$\forall i \in \{1, ..., k_n\}, n_i = \prod_{p \in S_i} p$$

Définition 3.3.4 *Un module n est dit **homogène** de niveau b si : $\forall p \in \mathbb{P}/p|n, h(p) = b$*

Remarque : Les sous-modules sont des modules homogènes.

Proposition 3.3.1 *Soit n un module composé de f facteurs.*
Soit k_n le nombre de niveaux différents des facteurs.
Tout module n est le produit de k_n modules homogènes.

3.3.2 Définitions sur les décompositions et la factorisation d'un entier

Définition 3.3.5 *On appelle **décomposition d'un entier** m, noté $D(m)$, le résultat d'une méthode d'extraction de 2 entiers tel que :*

$$\begin{array}{rccc} D: & \mathbb{N} & \longrightarrow & \mathbb{N}^2 \\ & m & \longmapsto & \{m_1, m_2\} \end{array}$$

où $m = m_1 m_2$
On note $\mathcal{L}(\mathbb{N}, \mathbb{N}^2)$ l'ensemble de ces fonctions.

Définition 3.3.6 *Soit m un entier composé de f nombres premiers.*

On appelle ***factorisation de l'entier*** *m le résultat d'une méthode d'extraction des f nombres premiers de m tel que :*

$$\begin{array}{rccc} DFact: & \mathbb{N} & \longrightarrow & \mathbb{P}^f \\ & m & \longmapsto & \{p_1, ..., p_f\} \end{array}$$

où $m = p_1 p_2 ... p_f$

Remarque : La factorisation d'un entier est aussi appelée la **décomposition totale** de cet entier. De la même façon, une décomposition d'un entier est appelée **factorisation partielle** de cet entier.

Définition 3.3.7 *Soit $p \in \mathbb{P}$ et $W_p = \{m \in \mathbb{N}/p|m\}$.*

On appelle ***décomposition d'un entier*** *m* ***ciblée sur*** *p, noté $D_p(m)$, le résultat de l'application :*

$$\begin{array}{rccc} D_p: & W_p & \longrightarrow & \mathbb{N}^2 \\ & m & \longmapsto & D(m) = \{m_1, m_2\} \end{array}$$

où $p = m_1$ ou $p = m_2$

Définition 3.3.8 *Soit $m \in \mathbb{N}$.*

Soit $\mathcal{L}(\mathbb{N}, \mathbb{N}^2)$ l'ensemble des fonctions définies par la Définition 3.3.5.

On appelle ***décomposition non triviale d'un entier*** *m une décomposition $D(m)$ qui vérifie $DNT_m(D) = 1$ où la fonction DNT_m est définie par :*

$$\begin{array}{rccl} DNT_m: & \mathcal{L}(\mathbb{N}, \mathbb{N}^2) & \longrightarrow & \{0, 1\} \\ & D & \longmapsto & 1 \text{ si } D(m) = \{m_1, m_2\} \text{ avec } m_1 \neq 1 \text{ et } m_2 \neq 1 \\ & & & 0 \text{ sinon} \end{array}$$

Remarque : Si $DNT_m(D) = 0$ alors D est une décomposition triviale de m.

Le problème mathématique de la factorisation partielle (et par conséquence totale) des grands nombres composés est basé sur l'hypothèse de la non-existence d'une Machine de Turing à temps polynomial en $|m|$ à laquelle on fournirait un entier m en entrée, et qui calculerait une décomposition $D(m)$ telle que $DNT_m(D) = 1$.

Définition 3.3.9 *Soit $m \in \mathbb{N}$ composé de plus de 2 nombres premiers.*

Soit $t \geq 2$.

Deux décompositions non triviales de m seront dites ***indépendantes*** *si les deux décompositions notées $D_{(1)}(m)$ et $D_{(2)}(m)$ vérifient $DNTI_m(D_{(1)}, D_{(2)}) = 1$,*

de même, plusieurs décompositions non triviales de m sont dites ***indépendantes*** *si elles sont indépendantes 2 à 2 : $DNTI_m(D_{(1)}, ...D_{(t)}) = 1$,*

où la fonction $DNTI_m$ est définie par :

$$\begin{array}{rccl} DNTI_m: & \mathcal{L}(\mathbb{N}, \mathbb{N}^2)^t & \longrightarrow & \{0, 1\} \\ & (D_{(1)}, ..., D_{(t)}) & \longmapsto & 1 \text{ si } \forall i \in \{1, ..., t\}, \forall j \neq i, D_{(i)}(m) \neq D_{(j)}(m) \\ & & & 0 \text{ sinon} \end{array}$$

En particulier, une ***décomposition non triviale*** D ***d'un entier*** m ***composé de 2 nombres premiers*** *constitue une décomposition indépendante de cet entier. Par extension de notation, on note* $DNT_m(D) = DNTI_m(D) = 1$.

Proposition 3.3.2 *Soit n un module composé de f nombres premiers.*

S'il existe un ensemble de décompositions non triviales du module n de cardinal supérieur à $f-1$ alors il existe un sous-ensemble de décompositions non triviales indépendantes.

$$\begin{array}{l}\forall(D_{(1)},\ldots,D_{(t)}) \in \mathcal{L}(\mathbb{N},\mathbb{N}^2)^t / \forall i : 1,\ldots,t, DNT_n(D_{(i)}) = 1,\\ card(\{D_{(1)}(n),...,D_{(t)}(n)\}) \geq f-1 \Rightarrow\\ \exists(D'_{(1)},\ldots,D'_{(f-1)}) \in \{D_{(1)},\ldots,D_{(t)}\}^{f-1} / DNTI_n(D'_{(1)},...,D'_{(f-1)}) = 1\end{array}$$

Proposition 3.3.3 *Soit n un module composé de f nombres premiers.*

S'il existe un ensemble de $f-1$ décompositions indépendantes du module n alors il existe un algorithme de factorisation du module n :

$$\forall(D_{(1)},\ldots,D_{(f-1)}) \in \mathcal{L}(\mathbb{N},\mathbb{N}^2)^{f-1} / DNTI_n(D_{(1)},...,D_{(f-1)}) = 1 \Rightarrow [FACT]$$

Théorème 3.3.1 *Soit n un module composé de f nombres premiers.*

Soit $\mathcal{L}(\mathbb{N},\mathbb{N}^2)$ l'ensemble des fonctions définies par la Définition 3.3.5.

S'il existe un ensemble de décompositions non triviales du module n de cardinal supérieur à $f-1$ alors il existe un algorithme de factorisation du module n :

$$\begin{array}{l}\forall(D_{(1)},\ldots,D_{(t)}) \in \mathcal{L}(\mathbb{N},\mathbb{N}^2)^t / \forall i : 1,\ldots,t, DNT_n(D_{(i)}) = 1,\\ card(\{D_{(1)}(n),...,D_{(t)}(n)\}) \geq f-1 \Rightarrow [FACT]\end{array}$$

Preuve. cf Propositions 3.3.2 et 3.3.3.

□

3.3.3 Conditions suffisantes de décomposition

Proposition 3.3.4 *Soit un module homogène n composé de f facteurs de niveau b.*

Pour tout élément ω du groupe $\mathbb{Z}_n^*$ tel que $\omega^{2^b} = 1 \bmod n$, s'il existe $1 \leq \alpha \leq b$ tel que $\omega^{2^\alpha} = 1 \bmod n$ et $\omega^{2^{\alpha-1}} \neq \pm 1 \bmod n$, alors on en déduit $D \in \mathcal{L}(\mathbb{N},\mathbb{N}^2)$ tel que $DNT_n(D) = 1$.

Preuve. On pose $x = \omega^{2^{\alpha-1}}$ et $y = 1$, et on applique le principe universel.

□

Proposition 3.3.5 *Soit un module homogène n composé de f facteurs de niveau b.*

Pour tout élément ω du groupe $\mathbb{Z}_n^*$ tel que $\omega^{2^b} = 1 \bmod n$, s'il existe $(i,j) \in \{1,...,f\}^2$ tels que $(p_i,p_j) \in S_b^2$ et $h_{p_i}(\omega) \neq h_{p_j}(\omega)$ alors on en déduit $D \in \mathcal{L}(\mathbb{N},\mathbb{N}^2)$ tel que $DNT_n(D) = 1$.

Preuve. Montrons que l'on vérifie les hypothèses de la Proposition 3.3.4 avec $\alpha = max(h_{p_i}(\omega), h_{p_j}(\omega))$.

Supposons, sans perte de généralité, que $\alpha = h_{p_i}(\omega)$. Comme on a $h_{p_i}(\omega) \neq h_{p_j}(\omega)$, cela signifie que $h_{p_i}(\omega) > h_{p_j}(\omega)$. Alors on a $\omega^{2^\alpha} = 1 \bmod p_i$ mais $\omega^{2^{\alpha-1}} = -1 \bmod p_i$, ainsi que $\omega^{2^\alpha} = 1 \bmod p_j$ et $\omega^{2^{\alpha-1}} = 1 \bmod p_j$. Donc $\omega^{2^\alpha} = 1 \bmod n$ et $\omega^{2^{\alpha-1}} \neq \pm 1 \bmod n$.

□

Remarque : On a

$$n = \prod_{p|n/h_p(g)<\alpha} p \times \prod_{p|n/h_p(g)\geq\alpha} p$$

Proposition 3.3.6 *Soit un module homogène n composé de f facteurs de niveau b.*

Pour toute paire de clés (g, Q) telle qu'il existe $(i, j) \in \{1, ..., f\}^2$ tels que $(p_i, p_j) \in S_b^2$ et $h_{p_i}(g) \neq h_{p_j}(g)$, on en déduit $D \in \mathcal{L}(\mathbb{N}, \mathbb{N}^2)$ tel que $DNT_n(D) = 1$.

Preuve. Soit p un facteur quelconque du module homogène n. Montrons que $h_p(\omega) = h_p(g)$ et que l'on se trouve donc dans le cas de la Proposition 3.3.5. Posons $\omega = Q^{2^k} g$. En appliquant la Proposition 3.2.6, on en déduit $h_p(\omega) \leq max(h_p(Q^{2^k}), h_p(g))$. Or $h_p(Q) = 0$ puisque Q est d'ordre impair, donc $h_p(\omega) = h_p(g)$.

□

Corollaire 3.3.1 *Soit un module homogène n composé de f facteurs de niveau b.*

Pour toute paire de clés (g, Q) telle qu'il existe $(i, j) \in \{1, ..., f\}^2$ tels que $(p_i, p_j) \in S_b^2$ et $(g|p_i) \neq (g|p_j)$, on en déduit $D \in \mathcal{L}(\mathbb{N}, \mathbb{N}^2)$ tel que $DNT_n(D) = 1$.

Preuve. Soit p un facteur quelconque du module homogène n. Montrons que l'on se trouve dans le cas de l'hypothèse de la Proposition 3.3.6.

Soit $g \in \mathbb{P} \cup \{2\}$. Supposons qu'il existe $(i, j) \in \{1, ..., f\}^2$ tels que $(p_i, p_j) \in S_b^2$ et $(g|p_i) \neq (g|p_j)$. Comme n est un module homogène alors $h_{p_i}(g) \neq h_{p_j}(g)$. En effet, sans perte de généralité, si $(g|p_i) = -1$ et $(g|p_j) = 1$, alors $h_{p_i}(g) = b$ et $h_{p_j}(g) < b$.

□

Proposition 3.3.7 *Selon les conditions suffisantes de l'hypothèse du Corollaire 3.3.1, la probabilité de dévoiler la factorisation d'un module homogène n de f facteurs à partir de m nombres de base choisis au hasard est minorée par la quantité $1 - \frac{f(f-1)}{2^{m+1}}$.*

Preuve. Si $(p_1, \ldots, p_f)$ sont les f facteurs du module n, et $(g_1, \ldots, g_m)$ sont les m nombres de base choisis aléatoirement selon une loi uniforme, on peut considérer les variables aléatoires suivantes $(X_{ik} = (g_k|p_i), i = 1, \ldots, f \text{ et } k = 1, \ldots, m)$ indépendantes de loi uniforme sur $\{-1, 1\}$.

On peut donc évaluer une borne inférieure de la probabilité recherchée en considérant la quantité :

$$Pr(\forall i, j \in \{1, \ldots, f\}, \exists k \in \{1, \ldots, m\} / \ X_{ik} \neq X_{jk}) \tag{3.1}$$

En introduisant la famille de vecteurs $\{\vec{Y_i} = (X_{ik} = (g_k|p_i)\ k = 1, \ldots, m)\,, i = 1, \ldots, f\}$ à valeurs dans $\{-1, 1\}^m$, on peut réécrire la probabilité (3.1) sous la forme suivante : $1 - Pr(\exists i \neq j \in \{1, \ldots, f\}\ \vec{Y_i} = \vec{Y_j})$

Pour expliquer la construction de la formule générale définissant $Pr(\exists i \neq j \in \{1, \ldots, f\}\ \vec{Y_i} = \vec{Y_j})$, on peut donner la factorisation de l'événement $[\exists i \neq j \in \{1, \ldots, f\}\ \vec{Y_i} = \vec{Y_j}]$ pour différentes valeurs de f :

Pour $f = 2$, on a $[\vec{Y_1} = \vec{Y_2}]$

Pour $f = 3$, on a $[\vec{Y_1} = \vec{Y_2}] \cup [[\vec{Y_1} \neq \vec{Y_2}] \cap [\vec{Y_3} = \{\vec{Y_1}, \vec{Y_2}\}]]$

Pour $f = 4$, on a $[\vec{Y_1} = \vec{Y_2}] \cup [[\vec{Y_1} \neq \vec{Y_2}] \cap [\vec{Y_3} = \{\vec{Y_1}, \vec{Y_2}\}]] \cup [[\vec{Y_1} \neq \vec{Y_2}] \cap [\vec{Y_3} = \{\vec{Y_1}, \vec{Y_2}\}] \cap [\vec{Y_4} = \{\vec{Y_1}, \vec{Y_2}, \vec{Y_3}\}]]$

Ces décompositions expliquent la formule générale suivante :

$$Pr(\exists i \neq j \in \{1, \ldots, f\}\ \vec{Y_i} = \vec{Y_j}) = \sum_{i=1}^{f-1} \frac{i}{2^m} \prod_{j=1}^{i-1} (1 - \frac{j}{2^m})$$

Par majoration de l'équation, on obtient :

$$\sum_{i=1}^{f-1} \frac{i}{2^m} \prod_{j=1}^{i-1} (1 - \frac{j}{2^m}) \leq \sum_{i=1}^{f-1} \frac{i}{2^m} \leq \frac{f(f-1)}{2^{m+1}}$$

□

Remarque :

Pour $f = 2$, la probabilité de dévoiler la factorisation à l'aide de m nombres de base est minorée par la quantité $1 - 1/2^m$.

Pour $f = 3$, la probabilité indique bien qu'on ne peut pas factoriser un module à 3 facteurs avec un seul nombre de base.

3.3.4 Techniques d'approximation par simulations

Les deux simulations suivantes permettent d'approximer la probabilité de trouver un ensemble de nombres de base qui engendre la factorisation de modules composés de f facteurs. Cet ensemble est recherché parmi les 54 premiers nombres premiers, et la notion de *moitié significative* est introduite pour recueillir un échantillon significatif des résultats.

Définition 3.3.10 *Soit n un module homogène. On définit G comme un ensemble de nombres de base engendrant des décompositions non triviales indépendantes du module n.*

Proposition 3.3.8 *(Simulation 1)*

Soit n un module quelconque composé de f facteurs.

Soit k_n le nombre de niveaux différents des facteurs.

Si, pour tout ensemble de niveau S_t du sous-module n_t du module n, il existe un ensemble d'éléments $G_t = \{g_{t,1}, ..., g_{t,s}\}$ défini par la Définition 3.3.10 telle que $s \geq |S_t|$, alors l'ensemble $G = \cup_{i \in \{1,\ldots,k_n\}} G_i$ est un ensemble de nombres de base engendrant des décompositions non triviales indépendantes du module n.

Voici la technique d'approximation n°1 :

On note S_t un ensemble de niveau relatif à un sous-module n_t de niveau t.

- $G = \varnothing$
- Pour tout t entre 1 et k_n
  - $G_t = \varnothing$
  - Pour tout $p_j \in S_t$, on recherche $D_{p_j}(n_t)$ selon le procédé suivant :
    - on recherche le plus petit nombre représenté sur un octet tel que $g \in \mathbb{P} \cup \{2\}$ noté g_{min} vérifiant :

$$(g|p_j) = -1 \; et \; \forall i / i \neq j \; et \; p_i \in S_t, (g|p_i) = 1$$

  - $G_t \leftarrow G_t \cup \{g_{min}\}$
  - $G \leftarrow G_t$

Dans le tableau suivant, on reporte les résultats obtenus par cette simulation sur la moitié significative des modules composés de 2, 3, 4 ou 5 facteurs. Le nombre de modules testés dans chacun des cas est de 10 000. Ainsi, on obtient le pourcentage de réussite pour chaque moitié significative à partir de la proportion du nombre d'échecs par rapport au nombre de cas considérés par cette moitié significative.

Listes de niveaux	Nombre de couples de facteurs testés	Nombre d'échecs
Moitié significative pour 2 facteurs : 56,25%		
[1,1]	2433	0
[2,1]	2567	0
[2,2]	6160	0
100% de réussite		
Moitié significative pour 3 facteurs : 57,42%		
[1,1,1]	1248	2
99,96% de réussite		
Moitié significative pour 4 facteurs : 50,24%		
[1,1,1,1]	601	88
[2,1,1,1]	1230	7
[2,2,2,1]	319	1
[2,2,2,2]	40	6
[3,1,1,1]	632	2
[3,3,3,1]	34	2
97,89% de réussite		
Moitié significative pour 5 facteurs : 51,29%		
[1,1,1,1,1]	324	224
[2,1,1,1,1]	782	100
[2,2,1,1,1]	798	1
[2,2,2,1,1]	385	2
[2,2,2,2,1]	100	21
[2,2,2,2,2]	3	1
[3,1,1,1,1]	392	55
[3,2,1,1,1]	753	1
[3,2,2,2,2]	23	4
92,08% de réussite		

Tab. 3.4 – Simulation Technique 1

Ces premiers résultats de simulation peuvent être améliorés en intégrant la probabilité plus importante de générer des facteurs de faible niveau. La recherche dans l'ensemble de niveau minimal du module peut donc être affinée, compte tenu de la probabilité plus forte d'échec de la recherche dans un ensemble de niveau de cardinal élevé.

Proposition 3.3.9 *(Simulation 2)*

Soit n un module quelconque composé de f facteurs.

Soit k_n le nombre de niveaux différents des facteurs.

Si, pour tout ensemble de niveau S_t du sous-module n_t du module n, il existe un ensemble d'éléments $G_t = \{g_{t,1}, ..., g_{t,s}\}$ défini par la Définition 3.3.10 tel que :

- *$s \geq |S_t|$ si $t \neq k_n$*
- *$s \geq |S_t| - 1$ si $t = k_n$*

alors l'ensemble $G = \cup_{i \in \{1,...,k_n\}} G_i$ est un ensemble de nombres de base engendrant des décompositions non triviales indépendantes du module n.

La technique d'approximation se décompose en 2 étapes, selon la nature de l'ensemble de niveau du module n_t considéré.

Soit S_t un ensemble de niveau relatif à un sous module n_t de niveau t.
$G = \varnothing$
- Si S_t est un ensemble de niveau non minimal
  $G_t = \varnothing$
  Pour tout $p_j \in S_t$, on recherche $D_{p_j}(n_t)$ selon le procédé suivant :
  - on recherche le plus petit nombre $g \in \mathbb{P} \cup \{2\}$ noté g_{min} vérifiant :

$$(g|p_j) = -1 \; et \; \forall i / i \neq j \; et \; p_i \in S_t, (g|p_i) = 1$$

  - $G_t \leftarrow G_t \cup \{g_{min}\}$
  - $G \leftarrow G \cup G_t$
- Si S_t est l'ensemble de niveau minimal
  - si $|S_t| = 1$ alors $G \leftarrow G$.
  - sinon,
    On fixe $p_t \in S_t$
    Pour tout $p_j \in S_t \backslash \{p_t\}$, on recherche $D_{p_j}(n_t)$ selon le procédé suivant :
    - on recherche le plus petit nombre $g \in \mathbb{P} \cup \{2\}$ noté g_{min} vérifiant :

$$(g|p_j) = -1 \; et \; \forall i / i \neq j \; et \; p_i \in S_t, (g|p_i) = 1$$

    - $G_t \leftarrow G_t \cup \{g_{min}\}$
      Si, pas de solution, on boucle avec un nouveau facteur $p_t \in S_t$
    - $G \leftarrow G \cup G_t$

Dans les mêmes conditions de simulation que précédemment, on reporte les résultats obtenus sur la moitié significative des modules composés de 2, 3, 4 ou 5 facteurs.

Listes de niveaux	Nombre de couples de facteurs testés	Nombre d'échecs
Moitié significative pour 2 facteurs : 56,25%		
[1,1]	2495	0
[2,1]	2543	0
[2,2]	6470	0
100% de réussite		
Moitié significative pour 3 facteurs : 57,42%		
[1,1,1]	1268	0
[2,2,2]	172	0
[3,3,3]	25	0
100% de réussite		
Moitié significative pour 4 facteurs : 50,24%		
[1,1,1,1]	639	3
[2,1,1,1]	1243	1
[2,2,2,1]	329	2
[3,3,3,1]	330	1
99,86% de réussite		
Moitié significative pour 5 facteurs : 51,29%		
[1,1,1,1,1]	312	69
[2,1,1,1,1]	793	5
[2,2,2,1,1]	406	5
[2,2,2,2,1]	109	11
[2,2,2,2,2]	10	2
[3,1,1,1,1]	392	3
[3,2,2,2,1]	207	1
[3,3,3,3,2]	5	2
98,09% de réussite		

TAB. 3.5 – Simulation Technique 2

Théorème 3.3.2 *Soit n un module quelconque composé de f facteurs.*

Soient $G_m = \{g_1, ..., g_m\}$ l'ensemble des nombres de base utilisés pour le protocole GQ2.

S'il existe un ensemble G vérifiant les conditions de la Proposition 3.3.8 ou de la Proposition 3.3.9 et si $G \subseteq G_m$, alors G_m est un ensemble de nombres de base permettant au protocole GQ2 d'être aussi sûr que le problème de la factorisation des grands nombres.

Preuve. cf Propositions 3.3.8 et 3.3.9, et Théorème 3.3.1

□

Bases pour l'estimation

Soit n un module composé de f facteurs et $b_1, b_2, ..., b_f$ les f niveaux respectifs des facteurs.

On définit une suite finie $h_1, h_2, ..., h_{k_n}$ des niveaux des facteurs, que l'on ordonne par ordre décroissant : $h_1 \geq h_2 \geq ... \geq h_{k_n}$.

Définition 3.3.11 *Soit l'ensemble E défini par*

$$E = \{(h_1, ..., h_f) \in \mathbb{N}^{*f} / p_1 \leftarrow \mathbb{P}(), ..., p_f \leftarrow \mathbb{P}(), \prod_{k=1}^{f} (\sum_{i=1}^{h_k} Pr(p \leftarrow \mathbb{P}() / h(p) = i)) > 0.5\}$$

Soit la norme euclidienne $N : (x_1, ..., x_f) \longmapsto \sqrt{x_1^2 + ... + x_f^2}$.
On définit le ***f-uplet minimum*** *noté* $(h_{1_{min}}, ..., h_{f_{min}})$ *par :*

$$(h_{1_{min}}, ..., h_{f_{min}}) = \{(h_1, ..., h_f) \in E / \forall (h'_1, ..., h'_f) \in E, N(h_1, ..., h_f) \leq N(h'_1, ..., h'_f)\}$$

Définition 3.3.12 *On appelle* ***moitié significative*** *des modules composés de f facteurs, les modules dont l'ensemble des niveaux des facteurs possède une norme euclidienne associée inférieure à celle du f-uplet minimum.*

f	2	3	4	5
$(h_{1_{min}}, ..., h_{f_{min}})$	(2,2)	(3,3,2)	(3,3,3,2)	(3,3,3,3,3)

TAB. 3.6 – Liste des f-uplets minima

La technique n°2 obtient de meilleurs résultats que la technique n°1 grâce à ses conditions de recherche plus affinées. Cependant, les résultats sur les moitiés significatives des modules composés de 2, 3, 4 ou 5 facteurs restent supérieurs à 9 chances sur 10 dans les deux cas.

3.3.5 Le choix optimum de petits nombres de base premiers

Nous justifions dans ce paragraphe le choix de prendre des nombres de base dans l'ensemble des nombres premiers. Pour cela, montrons qu'un nombre, composé par deux nombres premiers qui ne vérifient pas les conditions de l'hypothèse de la Proposition 3.3.6, a de très faibles chances de vérifier également ces conditions.

Proposition 3.3.10 *Soit n un module composé de deux facteurs* $p_1, p_2 \in \mathbb{P}$.
Pour tout élément $g \in \mathbb{Z}_n^*$ *tel que* $h_{p_1}(g) = h_{p_2}(g)$, *on a* $\forall i \in \mathbb{N}^*$, $h_{p_1}(g^{2^i}) = h_{p_2}(g^{2^i})$.

Preuve. cf Proposition 3.2.6

□

On en déduit qu'il est inutile de considérer les carrés successifs comme nombres de base.

Proposition 3.3.11 *Soit n un module composé de deux facteurs* $p_1, p_2 \in \mathbb{P}$ *et b le niveau de n.*

Soient $g_1, g_2 \in \mathbb{Z}_n^*$ *tels que* $h_{p_1}(g_1) = h_{p_2}(g_1)$ *et* $h_{p_1}(g_2) = h_{p_2}(g_2)$. *Alors* $h_{p_1}(g_1 g_2) = h_{p_2}(g_1 g_2)$ *avec une probabilité majorée par* $1 - 1/2^b$.

Preuve. Soient $g_1, g_2 \in \mathbb{Z}_n^*$ tels que $h_{p_1}(g_1) = h_{p_2}(g_1)$ et $h_{p_1}(g_2) = h_{p_2}(g_2)$.
Sous l'hypothèse que $h_{p_1}(g_1) \neq h_{p_1}(g_2)$, on a $h_{p_1}(g_1 g_2) = h_{p_2}(g_1 g_2) = max(h_{p_1}(g_1), h_{p_1}(g_2))$ d'après la Proposition 3.2.6. Or, la probabilité que $h_{p_1}(g_1) \neq h_{p_1}(g_2)$ est de $1 - 1/2^{min(b_{p_1}, b_{p_2})} < 1 - 1/2^b$.

□

On en déduit qu'il est donc peu intéressant en probabilité de considérer les nombres composés comme nombres de base.

On se base également sur l'hypothèse de Riemann qui annonce une forte probabilité de trouver des *petits* nombres non résidus quadratiques, pour limiter la recherche de nombres de base dans l'ensemble des petits nombres premiers.

3.4 Sécurité du protocole pour deux facteurs

Dans le chapitre précédent relatif à la preuve de sécurité du protocole GQ2, le Théorème 2.4.1 indique que le protocole GQ2 est une preuve zero-knowledge de connaissance de la factorisation du module si 2^{km} croît comme un polynôme en $|n|$, et si $\epsilon = 1/2^{km}$ où ϵ est le seuil de sécurité de la preuve d'équivalence avec la factorisation.

Dans le cas général de m nombres de base, une première estimation de la valeur de ϵ pour deux facteurs congrus à 3 modulo 4 est donnée dans la Proposition 2.4.4 : $\epsilon = 1 - 2^{-m}$. En fait, on démontre ici que $\epsilon = 1/2$, et on affine le résultat dans le cas général de facteurs quelconques en fonction des niveaux des facteurs : $1/2^b \leq \epsilon \leq 1/2$ si $k \geq b$. Ce dernier résultat est obtenu à partir d'une conjecture, et validé par simulation. La borne minimale sera atteinte pour $b_1 = 1$ (sans restriction de généralité) : la zone incompressible où l'on ne peut rien affirmer pour l'équivalence du protocole GQ2 avec la factorisation se voit, par ce résultat, réduite.
Un tableau récapitulatif des différents cas sera donné.

Notation : $\mathcal{H}$ est l'ensemble des éléments de $\mathbb{Z}_n^*$ qui vérifient les hypothèses de la Proposition 3.3.6.

3.4.1 Cas de facteurs congrus à 3 modulo 4

Dans le cas de modules composés de deux facteurs congrus à 3 modulo 4, le paramètre d'adaptation b est égal à 1 et on démontre que $\epsilon = 1/2$.

Lemme 3.4.1 *Soit n un module composé de deux facteurs congrus à 3 modulo 4.*
Soient $(g_1, \ldots, g_m)$ les m nombres de base du protocole GQ2 et soit $(e_1, \ldots, e_m) \in \{0, \ldots, 2^k - 1\}^m$.

S'il existe au moins un nombre de base g tel que $g \in \mathcal{H}$, alors $\prod_{i=1}^{m} g_i^{e_i} \in \mathcal{H}$ avec une probabilité de $1/2$, où les variables aléatoires e_i respectent une loi aléatoire uniforme sur $\{0, \ldots, 2^k - 1\}$.

Preuve. On suppose sans restriction de généralité que $g_1 \in \mathcal{H}$.

Si $\prod_{i=1}^{m} g_i^{e_i} \in \mathcal{H}$ alors on ne fait rien. Sinon, on a deux cas possibles : $g_1^{e_1} \in \mathcal{H}$ et $\prod_{i=2}^{m} g_i^{e_i} \in \mathcal{H}$, ou, $g_1^{e_1} \notin \mathcal{H}$ et $\prod_{i=2}^{m} g_i^{e_i} \notin \mathcal{H}$. Dans chacun des cas, en posant $e_1 = e_1 + 1 \bmod 2^k$ on a $\prod_{i=1}^{m} g_i^{e_i} \in \mathcal{H}$. En effet, si e_1 est impair alors $g_1^{e_1} \in \mathcal{H}$, et $g_1^{e_1} \notin \mathcal{H}$ sinon.

Ainsi, on peut diviser l'ensemble des défis possibles en deux sous-ensembles égaux D et $\bar{D}$ tel que $D = \{(e_1, \ldots, e_m) \in \{0, \ldots, 2^k - 1\}^m / \prod_{i=1}^{m} g_i^{e_i} \in \mathcal{H}\}$.

□

Proposition 3.4.1 *Soit n un module composé de deux facteurs congrus à 3 modulo 4. Soient $(g_1, \ldots, g_m)$ les m nombres de base du protocole GQ2, et soient $\{W, d, D\}$ et $\{W, d^*, D^*\}$ des triplets entrelacés.*

Alors $\{W, d, D\}$ et $\{W, d^*, D^*\}$ sont des triplets entrelacés valides avec une probabilité de $1/2$.

Preuve. Les triplets entrelacés donnent l'équation $(D^*/D)^{2^{k+1}} = \prod_{i=1}^{m} G_i^{d_i - d_i^*} \bmod n$. D'après le Lemme 3.4.1, la probabilité que $pgcd((D^*/D)^{2^k} - \prod_{i=1}^{m} g_i^{(d_i - d_i^*)}, n)$ soit égal à l'un des facteurs du module est de $1/2$.

□

Remarque 1 : ce résultat correspond à la probabilité de trouver un nombre $g \in \mathcal{H}$.

Remarque 2 : ce résultat dans le cas de deux facteurs congrus à 3 modulo 4, s'obtient grâce aux propriétés remarquables suivantes :
- $(g_1, g_2) \in \mathcal{H}^2 \Rightarrow g_1 g_2 \notin \mathcal{H}$
- $g_1 \in \mathcal{H}$ *et* $g_2 \notin \mathcal{H} \Rightarrow g_1 g_2 \in \mathcal{H}$
- $(g_1, g_2) \notin \mathcal{H}^2 \Rightarrow g_1 g_2 \notin \mathcal{H}$

Dans le cas de deux facteurs quelconques, on n'a plus les propriétés remarquables décrites ci-dessus. En effet, il existe des cas «pathologiques » qui ne permettent pas de raisonner de manière identique (cf Proposition 3.2.6) :
- $\exists (g_1, g_2) \in \mathcal{H}^2$ / $g_1 g_2 \in \mathcal{H}$ (si $max(h_{p_1}(g_1), h_{p_1}(g_2)) \neq max(h_{p_2}(g_1), h_{p_2}(g_2))$)
- $\exists (g_1, g_2) \notin \mathcal{H}^2$ / $g_1 g_2 \in \mathcal{H}$ (si $h_{p_1}(g_1) = h_{p_1}(g_2)$ et $h_{p_1}(g_1 g_2) \neq h_{p_2}(g_1 g_2)$)
- $\exists g_1 \in \mathcal{H}$ et $g_2 \notin \mathcal{H}$ / $g_1 g_2 \notin \mathcal{H}$ (si $h_{p_1}(g_2) > max(h_{p_1}(g_1), h_{p_2}(g_1))$)

3.4.2 Cas de facteurs quelconques

Dans le cas de modules composés de deux facteurs quelconques, on montre à partir d'une conjecture que $1/2^b \leq \epsilon \leq 1/2$ si $k \geq b$, où ϵ est le seuil de sécurité de la preuve d'équivalence avec la factorisation, et b le niveau du module.

Voici la conjecture suivante :

$$Pr((e_1,\ldots,e_m) \leftarrow \{0,\ldots,2^k-1\}^m / h_{p_1}(\prod_{i=1}^{m} g_i^{e^i}) \neq h_{p_2}(\prod_{i=1}^{m} g_i^{e^i})) = Pr(g \leftarrow \mathbb{P} \cup \{2\} / h_{p_1}(g) \neq h_{p_2}(g))$$

Application de la conjecture

Soit n un module composé de deux facteurs quelconques de niveaux b_1 et b_2.

Soit P_{echec} la probabilité définie par la Définition 3.2.3.

Soient $(g_1,\ldots,g_m)$ les m nombres de base du protocole GQ2 et soit $(e_1,\ldots,e_m) \in \{0,\ldots,2^k-1\}^m$.

S'il existe au moins un nombre de base g tel que $g \in \mathcal{H}$, alors $\prod_{i=1}^{m} g_i^{e_i} \in \mathcal{H}$ avec une probabilité égale à $1 - P_{echec}(b_1,b_2)$, où les e_i respectent une loi aléatoire uniforme sur $\{0,\ldots,2^k-1\}$.

Ce résultat est validé par simulation : tout se passe comme si on pouvait assimiler la variable $\prod_{i=1}^{m} g_i^{e_i}$ à la variable g. Ainsi la probabilité d'obtenir des triplets entrelacés valides correspond à la probabilité de trouver un nombre $g \in \mathcal{H}$:

$$1 - P_{echec}(b_1,b_2) = 1 - \frac{2}{2^{b_1+b_2}} - \sum_{i=2}^{min(b_1,b_2)} \frac{(2^{i-1})^2}{2^{b_1+b_2}}.$$

- Si $b_1 = b_2 = 1$ alors $1 - P_{echec}(b_1,b_2) = 1/2$
- Si $b_2 = 1$ alors $1 - P_{echec}(b_1,b_2) = 1 - 1/2^{b_1}$
- Sinon $1 - P_{echec}(b_1,b_2) = 1 - \frac{2}{2^{b_1+b_2} \times 3} - \frac{1}{2^{b_1-b_2} \times 3}$

Intuitivement, on explique ce résultat en faisant des approximations à partir de la preuve pour les facteurs congrus à 3 modulo 4. En ne considérant pas les cas «pathologiques », on remarque ensuite que le cas où $g^e \in \mathcal{H}$ est supérieur à 1 chance sur 2 en probabilité, puisque certains défis pairs peuvent convenir, en plus des défis impairs.

Lemme 3.4.2 *Soient b_1, b_2 deux entiers strictement positifs tels que $b_1 \geq b_2$.*
La probabilité $1 - P_{echec}(b_1,b_2)$ est majorée par $1 - 1/2^{b_1}$.

Preuve. $1 - \frac{1}{2^{b_1}} - (1 - P_{echec}(b_1,b_2)) = \frac{1}{2^{b_1} \times 3}(-3 + 2^{1-b_2} + 2^{b_2}) \geq 0$

□

Intuitivement, on voit que plus la différence entre les niveaux des facteurs est grande, plus la probabilité $1 - P_{echec}(b_1,b_2)$ est grande, et, est maximale lorsque le niveau le plus bas est de plus égal à 1.

Tous triplets entrelacés $\{W,d,D\}$ et $\{W,d^*,D^*\}$ sont des triplets entrelacés valides avec une probabilité de $1 - P_{echec}$ si $k \geq b$.

En effet, les triplets entrelacés donnent l'équation $(D^*/D)^{2^{k+b}} = \prod_{i=1}^{m} G_i^{d_i - d_i^*} \bmod n$.
D'après la conjecture, et dans le cas où $k \geq b$, la probabilité qu'il existe $j \in \{0,\ldots,b-1\}$

tel que $pgcd((D^*/D)^{2^{k+j}} - \prod_{i=1}^{m} g_i^{(d_i - d_i^*)2^j}, n)$ soit égal à l'un des facteurs du module est de $1 - P_{echec}$.

Dans le cas $b_2 = 1$, c'est-à-dire dans la moitié des cas de modules composés de deux facteurs, GQ2 est une preuve de connaissance de la factorisation lorsque $m = 1$ et $k = b$. Sinon, quels que soient les paramètres k et m, on ne peut déduire l'équivalence avec la factorisation seulement quand l'attaquant a une probabilité de s'authentifier supérieure à $1 - 1/2^b$. La zone incompressible où l'on ne peut rien déduire s'est cependant réduite par rapport au résultat donné dans le Chapitre 2.

3.4.3 Tableau récapitulatif

Selon les cas de facteurs congrus à 3 modulo 4 ou quelconques, les conditions de sécurité optimale pour déduire l'équivalence du protocole GQ2 avec la factorisation des grands nombres se déduisent de la vérification de l'égalité du seuil de sécurité ϵ, avec la probabilité minimale d'usurpation d'identité.

$k \geq b$	Borne à atteindre	$p, q = 3$ *modulo* 4	p *ou* q = 3 *modulo* 4	p, q *quelconques*
	$1/2^{km}$	$\epsilon = 1/2$	$\epsilon = 1/2^b$	$1/2^b \leq \epsilon \leq 1/2$

TAB. 3.7 – Récapitulatif des valeurs du seuil de sécurité

3.5 Extension à de plus larges exposants publics

Cette partie met en avant le raisonnement identique que l'on peut appliquer au protocole GQ2 pour un exposant public v de la forme e^{k+b} où $e = 2, 3, 4, 5, 7, 11$: on raisonne sur des anneaux isomorphes à des anneaux euclidiens connus (cf Théorème 1.2.10). En effet, les calculs du symbole de Legendre, de Jacobi et celui du pgcd de deux entiers qui utilisent l'algorithme d'Euclide, doivent subsister dans les nouveaux espaces considérés. On généralise ainsi l'isomorphisme décrit dans la Proposition 1.2.8 à de plus larges exposants.

3.5.1 Définitions et lemmes

Définition 3.5.1 *Par la suite, on appelle* ***nombre premier de*** 1^{ere} ***espèce pour la puissance*** *e, un nombre p premier tel que* $p = 1 \bmod e$ *et tel que* $p - 1$ *s'écrit sous la forme* $p - 1 = e^{b_p} p_1$ *avec* $pgcd(p_1, e) = 1$.

Définition 3.5.2 *Soit p un nombre premier de* 1^{ere} *espèce pour la puissance e. On généralise l'appellation de* ***niveau de*** *p pour la puissance e noté* b_p, *où* $p - 1 = e^{b_p} p_1$ *avec* $pgcd(p_1, e) = 1$.

Définition 3.5.3 *Soit p un nombre premier de* 1^{ere} *espèce pour la puissance e, et soit* b_p *le niveau de p pour la puissance e tel que* $p - 1 = e^{b_p} p_1$. *L'application* η_p *pour la puissance e est généralisée par :*

$$\begin{array}{rccc} \eta_p : & \mathbb{Z}_p^* & \longrightarrow & \mathbb{Z}_p^* \\ & x & \longmapsto & x^{p_1} \bmod p \end{array}$$

Définition 3.5.4 *Soit p un nombre premier de 1^{ere} espèce pour la puissance e, et g un élément du groupe multiplicatif $\mathbb{Z}_p^*$.*

On appelle ***niveau de l'élément*** *g le nombre $h_p(g)$ tel que :*

$$h_p(g) = min\{i \in \mathbb{N} / \eta_p(g)^{e^i} = 1 \bmod p\}$$

Définition 3.5.5 *Pour tout nombre premier p de 1^{ere} espèce pour la puissance e et a un entier tel que $pgcd(a,p) = 1$, le symbole de Legendre généralisé $(a|p)_e$ (appelé caractère quadratique, cubique ou biquadratique pour e=2, 3 ou 4) est défini pour être égal à 1 si a est un résidu pour la puissance e modulo p, et différent de 1 sinon. Pour être complet, on définit $(a|p)_e = 0$ si p divise a.*

Proposition 3.5.1 *Soit p un nombre premier de 1^{ere} espèce pour la puissance e et soit $\alpha \in \mathbb{Z}$. Alors on a : $(\alpha|p)_e = \alpha^{\frac{p-1}{e}} \bmod p$.*

Proposition 3.5.2 *Soit un module homogène n de niveau b composé de f facteurs de 1^{ere} espèce pour la puissance e.*

Pour tout élément ω du groupe $\mathbb{Z}_n^*$ tel que $\omega^{e^b} = 1 \bmod n$, s'il existe $1 \leq \alpha \leq b$ tel que $\omega^{e^\alpha} = 1 \bmod n$ et $\omega^{e^{\alpha-1}} \neq \pm 1 \bmod n$ alors on en déduit $D \in \mathcal{L}(\mathbb{N}, \mathbb{N}^2)$ telle que $DNT_n(D) = 1$.

Preuve. On pose $x = \omega^{e^{\alpha-1}}$ et $y = 1$. En généralisant le principe universel à la puissance e, on a $x^e = y^e \bmod n$ et $x \neq \pm y \bmod n$,
donc $n = pgcd(x-y, n) \times pgcd((x^e - y^e)/(x-y), n)$.

□

Proposition 3.5.3 *Soit un module homogène n de niveau b composé de f facteurs de 1^{ere} espèce pour la puissance e.*

Pour tout élément ω du groupe $\mathbb{Z}_n^*$ tel que $\omega^{e^b} = 1 \bmod n$, s'il existe $(i,j) \in \{1,...,f\}^2$ tels que $(p_i, p_j) \in S_b^2$ et $h_{p_i}(\omega) \neq h_{p_j}(\omega)$ alors on en déduit $D \in \mathcal{L}(\mathbb{N}, \mathbb{N}^2)$ tel que $DNT_n(D) = 1$.

Proposition 3.5.4 *Soit un module homogène n de niveau b composé de f facteurs de 1^{ere} espèce pour la puissance e.*

Pour toute paire de clés (g, Q) telle qu'il existe $(i,j) \in \{1,...,f\}^2$ tels que $(p_i, p_j) \in S_b^2$ et $h_{p_i}(g) \neq h_{p_j}(g)$, on en déduit $D \in \mathcal{L}(\mathbb{N}, \mathbb{N}^2)$ tel que $DNT_n(D) = 1$.

Corollaire 3.5.1 *Soit un module homogène n de niveau b composé de f facteurs de 1^{ere} espèce pour la puissance e.*

Pour toute paire de clés (g, Q) telle qu'il existe $(i,j) \in \{1,...,f\}^2$ tels que $(p_i, p_j) \in S_b^2$ et $(g|p_i)_e = 1$ et $(g|p_j)_e \neq 1$, on en déduit $D \in \mathcal{L}(\mathbb{N}, \mathbb{N}^2)$ tel que $DNT_n(D) = 1$.

3.5.2 Protocole généralisé

Elaboration des bi-clés GQ2

Soit b le niveau du groupe multiplicatif $\mathbb{Z}_n^*$.
Soit e le paramètre prenant les valeurs 2, 3, 4, 5, 7, 11 selon les cas.

Soit ω_i tel que $\omega_i = Q_i^{e^k} g_i$ pour chaque nombre de base.

Les m équations génériques sont alors $\omega_i^{e^b} = 1 \bmod n$ car $v = e^{k+b}$.

La construction d'une bi-clé GQ2 s'effectue selon les étapes suivantes :
- On choisit aléatoirement deux nombres premiers rationnels p_1 et p_2 de 1^{ere} espèce pour la puissance e
- On calcule le module n égal au produit de p_1 par p_2
- La clé publique GQ2 se compose du module n et de m nombres publics notés $(G_1, ..., G_m)$ chaque G_i étant la puissance $e^{b\ ieme}$ d'un petit nombre premier rationnel noté g_i : $\forall i \in \{1, ..., m\},\ G_i = g_i^{e^b}$
- La clé publique GQ2 doit vérifier la propriété suivante :
  Pour au moins un nombre de base, noté g, nous avons : $(g|p_1)_e = 1$ et $(g|p_2)_e \neq 1$ si $b_{p_1} = b_{p_2}$ et $(g|p_1)_e \neq 1$ si $b_{p_1} > b_{p_2}$ sans perte de généralité
- La clé privée GQ2 se compose des nombres premiers p_1, p_2 et de m nombres secrets notés $(Q_1, ..., Q_m)$ reliés aux nombres publics par les équations génériques suivantes :

$$\forall i \in \{1, ..., m\}, G_i Q_i^{e^{k+b}} = 1 \bmod n$$

Alice		Bob
Clé privée : (p_1, p_2, Q_i)		Clé publique : (n, b, k, g_i, e)
$r \in \mathbb{Z}_n^*$ $W = r^{e^{k+b}} \bmod n$	$\longrightarrow$	
	$\longleftarrow$	$d_1, ..., d_m \in \{0, ..., e^k - 1\}$
$D = rQ_1^{d_1} ... Q_m^{d_m} \bmod n$	$\longrightarrow$	
		$W = ?D^v (g_1^{e^b})^{d_1} ... (g_m^{e^b})^{d_m} \bmod n$

FIG. 3.2 – Schéma du protocole GQ2 pour $v = e^{k+b}$

3.6 Complexité du protocole étendu

Nous regardons l'impact de la variation des différents paramètres introduits dans le protocole GQ2 généralisé. La référence prise pour comparer les performances est le protocole de Fiat-Shamir. Deux niveaux de sécurité seront pris comme exemple.

3.6.1 Complexités incluant tous les paramètres

L'étude de la variation de tous les paramètres du protocole GQ2 permet d'avoir une idée générale des performances selon les situations : on exprime les complexités CPC, CPV, CS, CM en fonction de k, m, b, f, e et de la variable $coeff$, qui dépend de

e et correspond au ratio considéré entre la complexité d'un carré (resp un cube, une puissance quatrième) modulaire et une multiplication modulaire.

Les précalculs $CPC(A), CPV(A), CS(A), CM(A)$ ne prennent pas en compte le paramètre f dans un premier temps (cf Tableau 3.8). La variable X est introduite pour la technique des restes chinois (cf Partie 2.3.1).

Complexité	GQ2 généralisé
CPC(A)	$(k+b) \times coeff + (k-1) \times coeff + k \times m/2$
CPC	$(2 \times X(f) + f \times CPC(A))/f^2$
CPV(A)	$(k+b) \times coeff$
CPV	CPV(A)
CS(A)	$m \times \|n\|$
CS (Kbits)	$CS(A) + (2 \times f - 1) \times \|n\|/f$
CM(A)	$\|h\| + \|n\| + k \times m/1024$
CM (Kbits)	CM(A)

TAB. 3.8 – Complexité GQ2 généralisée

3.6.2 Performances pour une authentification faible et forte

Nous comparons numériquement ces résultats de complexité avec le schéma de Fiat-Shamir. Dans les annexes, des diagrammes illustrent ces résultats (cf Annexes A.2) : les colonnes pleines signifient que le protocole GQ2 reste plus performant que ce dernier avec les paramètres considérés. Une **authentification faible** correspond à un niveau de sécurité de 2^{-16} pour chaque tour d'exécution du protocole, et pour une **authentification forte**, le niveau de sécurité est fixé à 2^{-36} : on reprend ces appellations introduites dans la norme ISO/IEC 9798 [ISO04]. Côté prouveur et vérifieur, le nombre de multiplications modulaires pour le schéma de Fiat Shamir est de 11 pour une authentification faible, et de 22.5 pour une authentification forte. Les paramètres considérés pour le schéma GQ2 sont $k = 4, m = 4$ dans le premier cas, et $k = 6, m = 6$ dans le deuxième.

Le théorème suivant permet d'approximer la probabilité de générer un module composé de f facteurs, de niveau b suffisamment grand, en particulier quand $e = 2$. On en déduit ainsi l'ordre de grandeur du risque que les temps de calculs côté prouveur et vérifieur du protocole GQ2, dépassent ceux du schéma de Fiat-Shamir.

Lemme 3.6.1 *Soit $i \in \mathbb{N}^*$,*

$$Pr(p \leftarrow \mathbb{P}(), b_p = i) = \frac{1}{e^i} \text{ et } Pr(p \leftarrow \mathbb{P}(), b_p < i) = \sum_{j=1}^{i-1} \frac{1}{e^j} = \left(1 - \frac{1}{e^{i-1}}\right) \frac{1}{e-1}$$

Théorème 3.6.1 *Soit un module de f facteurs, la probabilité que le niveau du module soit de niveau b peut être approximé par $f/2^b$, pour b suffisamment grand.*

Preuve.
Soit f le nombre de facteurs qui composent le module.

Soit $P'_f(b)$ la probabilité que le niveau du module composé de f facteurs soit de niveau supérieur ou égal à b.

$$\begin{aligned} P'_f(b) &= Pr(\forall i \in \{1,...,f\}, p_i \leftarrow \mathbb{P}(), \exists i \in \{1,...,f\}, h(p_i) \geq b) \\ &= 1 - Pr(\forall i \in \{1,...,f\}, p_i \leftarrow \mathbb{P}(), h(p_i) < b) \\ &= 1 - \left(1 - \frac{1}{e^{b-1}}\right)^f \left(\frac{1}{e-1}\right)^f \end{aligned}$$

Soit $P_f(b)$ la probabilité que le niveau du module composé de f facteurs soit de niveau b : $P_f(b) = P'_f(b) - P'_f(b+1)$

Or, pour $e = 2$, $P'_f(b) = 1 - \left(1 - \frac{1}{2^{b-1}}\right)^f$, et pour X petit, on a $1-(1-X)^f \sim fX$

Donc, pour toute donnée f fixée et $e = 2$, $P_f(b)$ peut-être approximé par $\frac{f}{2^b}$ pour b suffisamment grand.

□

A partir de ce résultat d'estimation et des complexités obtenues du schéma GQ2 avec l'ensemble de ses paramètres, le Tableau 3.9 donne l'ordre de grandeur du risque d'obtenir une complexité (CPC ou CPV) supérieure au schéma de Fiat-Shamir : le deuxième schéma le plus performant décrit dans la partie de la norme ISO/IEC 9798-5. On fixe ici $e = 2$ et $coeff = 0.75$. Pour une valeur de f fixée, on recherche le plus petit nombre b, noté b_0, tel que le temps de calculs du protocole GQ2 devient supérieur à celui de Fiat-Shamir : la probabilité de générer un module de niveau égal ou supérieur à b_0 est approximée par $f/2^{b_0-1}$.

	Risque pour auth faible				Risque pour auth forte			
f	2	3	4	5	2	3	4	5
CPC	2^{-8}	2^{-21}	2^{-32}	2^{-107}	2^{-22}	2^{-50}	2^{-76}	2^{-243}
CPV	2^{-9}	2^{-9}	2^{-8}	2^{-8}	2^{-27}	2^{-27}	2^{-26}	2^{-26}

TAB. 3.9 – Risques de performances GQ2 inférieures à celles de Fiat-Shamir

Chapitre 4

Signature de Rabin-Williams généralisée

La famille des schémas de signature électronique de type Rabin-Williams, initiée par M.O. Rabin en 1979, est basée sur le problème difficile de calcul de racines carrées. Ce chapitre est issu d'un travail conjoint avec Francois Arnault, et un projet d'article est en cours avec la collaboration de Benoît Libert et Jean-Jacques Quisquater.

Nous rappelons les différents concepts généraux introduits pour sécuriser ce type de schéma. Nous évoquons les points communs et les différences des améliorations proposées pour la signature de Rabin-Williams entre 1980 et 1999, dont l'intégration et la complexité sont intéressantes dans le cas de facteurs congrus à 3 modulo 4.

Dans ce chapitre, nous proposons un nouveau schéma qui regroupe les points positifs de ces schémas de type Rabin-Williams : grâce au maintien d'un exposant privé de signature, on applique un algorithme déterministe au lieu d'un algorithme probabiliste, quel que soit le module considéré, contrairement aux schémas élaborés ces dernières années. Les performances s'en trouvent améliorées et les étapes de calculs simplifiées. La preuve de sécurité de ce schéma se déduit aisément de celle de Rabin-Williams, et la généralisation à de plus larges exposants peut s'appliquer.

Dans la suite, nous étendrons les preuves connues sur l'apport limité de l'information transmise sur les niveaux des deux facteurs du module, induite dans les deux protocoles de ce document, GQ2 et Rabin-Williams.

4.1 Concept et sécurité sur la signature électronique

La signature électronique requiert des contraintes fortes sur la sécurité, compte tenu des enjeux qui en découlent sur une durée prolongée dans le temps. Les diverses techniques de falsification suivent des techniques d'attaques plus ou moins élaborées. Ainsi, les protections appliquées afin de prévenir une atteinte à l'intégrité ou l'authenticité du message sont rappelées ; en particulier dans le cas de schémas de signature prouvés équivalents à la factorisation de type Rabin-Williams. Quelques attaques connues sur ces schémas sont également évoquées.

4.1.1 Définitions

La **signature électronique** remplit les mêmes fonctions que la signature manuelle : elle engage la responsabilité d'un individu vis à vis d'un tiers. De ce fait, elle doit répondre aux mêmes exigences de sécurité : les propriétés d'intégrité et d'authenticité doivent être respectées. Il faut s'assurer que toute signature ne peut être ni réutilisable, ni forgeable : chaque signature doit être dans la capacité de maintenir sa dépendance avec le message signé et le signataire. Ainsi, le caractère **non répudiable** de la signature est respecté : personne ne pourra nier son acte de signature par la suite, et cette preuve électronique pourra être présentée devant un tribunal en cas de litige entre les deux parties.

4.1.2 Attaques et falsification

Attaques

Les **attaques** sur les signatures peuvent être de deux types : la tentative d'un tiers à engager frauduleusement la responsabilité d'un tiers en modifiant les données d'un message signé (intégrité), et celle à usurper l'identité d'un tiers (authenticité).

Dans le deuxième cas d'attaque, on peut distinguer une attaque passive d'une attaque active. Une **attaque passive** consiste en l'observation «muette » de la communication entre Alice et Bob. Les protocoles de sécurité où le **rejeu** est de mise, sont sensibles à ce type d'attaque : utilisation d'un mot de passe ou d'une valeur de signature (Yes-cards). Une **attaque active** se déroule, elle, en deux phases. Dans une première phase, l'attaquant communique avec Alice afin d'obtenir le maximum d'information lors de ces échanges, puis il usurpe l'identité d'Alice dans une deuxième phase.

Idéalement, ces attaques actives ont pour objectif de récupérer des informations sur la clé privée qu'Alice utilise pour signer, afin de pouvoir prendre sa place par la suite, et signer en son nom. Différentes attaques classées par ordre de sévérité dans le Tableau 4.1, peuvent être répertoriées, selon les conditions et les degrés de liberté laissés à l'attaquant durant la première phase de l'attaque.

L'attaque sans message est bien évidemment l'attaque la plus difficile (dans l'absolu). Pour les protocoles dont la sécurité est basée sur la factorisation du module, elle correspond à factoriser le module, c'est à dire à résoudre le problème difficile de la factorisation des grands nombres.

Falsification

Le degré de **falsification** d'un attaquant correspond à sa capacité d'attaque lors de la deuxième phase de l'attaque active : son pouvoir de nuisance envers autrui. Voici le classement de ces falsifications :

- Le **cassage total** : le niveau le plus fort de succès. L'attaquant retrouve la clé secrète.
- La **falsification universelle** : l'attaquant ne retrouve pas la clé secrète, mais il peut forger la signature de n'importe quel message

	Prise de connaissance de la clé publique		Couples (messages/signatures) étudiés		
Attaque...	avant les ensembles de couples	après les ensembles de couples	non choisis non adaptés	choisis non adaptés	choisis adaptés
...sans messages	⋆				
...à messages connus	⋆		⋆		
...à messages choisis générique		⋆		⋆	
...à messages choisis orientée	⋆			⋆	
...à messages choisis dynamique ou adaptative	⋆				⋆

TAB. 4.1 – Types d'attaques

- La **falsification sélective** : l'attaquant peut forger la signature de quelques messages.
- La **falsification existentielle** : l'attaquant peut forger une signature mais n'a aucun contrôle sur le message dont il obtient la signature.

Le cassage total est bien évidemment le pouvoir le plus élevé que puisse espérer un attaquant : il ne peut pas espérer plus.

Définition 4.1.1 *Un schéma de signature sera dit **sûr** s'il contre une falsification existentielle, dans le contexte d'attaque à messages choisis adaptative.*

Dans le cas des schémas de signature de type Rabin auxquels nous nous intéressons par la suite, leur **propriété de multiplicativité** entraîne une fragilité de ces schémas dans le cas d'attaque à messages choisis. Une falsification sélective est possible. Si on considère la fonction de signature S que le signataire applique au message M et M', l'attaquant peut signer le message $M \times M'$ en appliquant la propriété de multiplicativité $S(M \times M') = S(M) \times S(M')$, inhérente à chaque protocole dont la sécurité est prouvée équivalente à la factorisation.

La solution connue contre ce type d'attaque est l'application de «padding » de sécurité aux messages à signer.

4.1.3 Protections appliquées

De manière générale, la structure d'une signature électronique définie par S. Goldwasser, S. Micali et R.L. Rivest [GMR88] puis M. Bellare et P. Rogaway [BR96]), suit les 3 étapes suivantes :

Définition 4.1.2 *Un schéma de signature est constitué :*

- *d'un algorithme de* ***génération de clés****, qui est un algorithme probabiliste qui prend en entrée un paramètre de sécurité k et retourne le couple (pk, sk) de clés publique et privée.*
- *d'un algorithme de* ***signature*** *noté $Sign()$ qui, étant donné en entrée un message et un couple (pk, sk) de clés publique et privée, produit une signature.*
- *d'un algorithme de* ***vérification*** *noté $Verif()$ qui, étant donné une signature s, un message m et une clé publique pk, teste si s est une signature valide de m par rapport à pk.*

A ce schéma de base vont venir s'appliquer des mécanismes de format tels que les fonctions de redondance, les fonctions de hachage, FDH ou PSS. Ces mécanismes vont jouer un rôle essentiel de protection contre les attaques sur les schémas comme RSA, Rabin ou Rabin-Williams.

Fonction de redondance

Une description plus détaillée des différents types de fonctions de redondance est répertoriée dans le travail de synthèse de J.F. Misarsky [Mis98].

Définition 4.1.3 *Dans un schéma de signature, une* ***fonction de redondance*** *R est une fonction inversible, appliquée au message que l'on désire signer. L'inverse de cette fonction doit être facilement calculable.*

Dans la suite, on note M le message à signer.

Les **redondances affines** sont définies par $R(M) = w.M + a$ où w est appelé la **redondance multiplicative** et a **la redondance additive**. La **redondance fixée à gauche ou à droite** en est un cas particulier, et correspond à la concaténation du message avec une séquence de bits.

Cette précaution permet de réduire la possibilité de trouver deux messages possédant à la fois la même signature et une séquence de bits de format identique.

Une **redondance modulaire** consiste à appliquer une réduction modulaire au message, modulo un nombre fixé.

Dans le cas du schéma de Rabin-Williams décrit dans [MOV97], elle permet au vérifieur d'en déduire le message signé (parmi 4) à l'aide du résultat de cette réduction modulaire.

Fonction de hachage

Définition 4.1.4 *Une* ***fonction de hachage*** *est une fonction h définie de $\{0,1\}^*$ dans $\{0,1\}^n$ qui prend en entrée une séquence de bits de taille arbitrairement longue, et renvoie en sortie une séquence de bits de taille fixée.*

Une fonction de hachage est une fonction publique qui ne prend en entrée que le message considéré. Le caractère pseudo-aléatoire de ce haché permet de se protéger contre les attaques basées sur la propriété de multiplicativité des schémas. Une falsification existentielle n'est également plus possible grâce au sens unique de cette fonction.

Deux générations de fonctions de hachage se sont développées dans les années 90. La fonction de hachage MD4 (128 bits) présentée par Rivest en 1991[Riv91] a été à l'origine de la construction de MD5 [Riv92] et RIPEMD qui retournent un haché sur 128 bits, puis RIPEMD-160 [DBP96]. L'institut américain NIST met en place de son côté, SHA-0 en 1993 et SHA-1 en 1995. Aujourd'hui, MD5 et SHA-1 restent les deux fonctions de hachage les plus répandues.

Afin de ne pas affaiblir la sécurité du procédé de signature, les fonctions candidates à cette appellation de «fonction de hachage » doivent être à collisions faibles et fortes difficiles :

Définition 4.1.5 *Une fonction de hachage est à* ***collisions faibles difficiles*** *si, étant donné un message M, il est calculatoirement difficile d'obtenir un message $M' \neq M$ tel que $h(M') = h(M)$.*

Définition 4.1.6 *Une fonction de hachage est à* ***collisions fortes difficiles*** *s'il est calculatoirement difficile d'obtenir deux messages différents M' et M tels que $h(M') = h(M)$.*

Dans le cas contraire, un attaquant pourrait porter préjudice au signataire, en détenant une signature d'un message que le supposé signataire n'aurait pas signé (une falsification existentielle).

Récemment, la rump session de CRYPTO'04 a révélé les résultats d'une équipe chinoise [WLF+05]. Des attaques par collisions avec de fortes probabilités à l'aide de faibles ressources, ont été effectuées sur MD4 et RIPEMD, ainsi que MD5 [WY05]. Le SHA-0 [BCJ+05] dans sa version réduite, est également touché par cette cryptanalyse.

Fonction FDH

Ce schéma introduit en 1993 par M. Bellare et P. Rogaway [BR93], admet une preuve de sécurité dans le modèle de l'oracle aléatoire, et est basé sur le problème mathématique difficile RSA.

Définition 4.1.7 *Le schéma de signature FDH est défini par les 3 étapes suivantes :*

- *GenFDH(1^k)*
  $(pk, sk) \leftarrow \mathcal{RSA}(1^k)$
  Retourner (pk, sk)
- *SignFDH(M)*
  $y \leftarrow H_{FDH}(M)$
  Retourner $s \leftarrow Sign(y, pk, sk)$
- *VerifFDH* (M, s)
  $y \leftarrow Verif(s, M, pk)$
  $y' \leftarrow H_{FDH}(M)$
  Si $y = y'$ *alors* ***Retourner*** *1* ***sinon Retourner*** *0*

où $Sign()$ et $Verif()$ sont les fonctions de signature et de vérification du protocole RSA, et H_{FDH} une fonction de hachage dont la longueur du résultat est égale à la longueur de la clé.

Définition 4.1.8 *Une signature est dit de type hash-et-signe si l'algorithme de génération de signature commence par hacher le message pour ensuite signer le haché en utilisant la clé privée.*

Le schéma FDH est une signature de type hash-et-signe. La fonction PSS en est un autre exemple.

Fonction PSS

Le mécanisme de format PSS est une combinaison de 3 techniques : l'utilisation d'une fonction de hachage, d'une fonction de redondance et de l'introduction d'un nombre aléatoire qui rend l'algorithme de signature probabiliste. Introduit par M. Bellare et P. Rogaway [BR96], PSS est utilisé dans la norme ISO/IEC 14888-2 [ISOre] en cours de rédaction.

Soient $Sign()$ et $Verif()$ les algorithmes de signature et de vérification. On pose $k = 1024$, $k_0 = k_1 = 128$.

Soit M le message à signer.

- GenPSS(1^k)
  $(pk, sk) \leftarrow \mathcal{RSA}(1^k)$
  Retourner (pk, sk)
- SignPSS(M)
  $r \leftarrow \{0,1\}^{k_0}; w \leftarrow h(M||r); r^* \leftarrow G_1(w) \oplus r$
  $y \leftarrow 0||w||r^*||G_2(w)$
  Retourner $y \leftarrow Sign(y, pk, sk)$
- VerifPSS (M, s)
  $y \leftarrow Verif(s, M, pk)$
  Décomposer y tel que $b||w||r^*||\gamma$ (b le premier bit de y, w les k_1 bits suivants, r^* les k_0 bits suivants, et γ les bits restants)
  $r \leftarrow r^* \oplus G_1(w)$
  Si $(h(M||r) = w$ et $G_2(w) = \gamma$ et $b = 0)$ alors **Retourner** 1 **sinon Retourner** 0

où h et G sont deux fonctions de hachage à valeurs respectivement dans $\{0,1\}^{k_1}$ et $\{0,1\}^{k-k_1-1}$ (G_1 retourne les k_0 premiers bits du résultat de G et G_2 retourne les $k - k_0 - k_1 - 1$ bits suivants), r une valeur aléatoire.

La preuve de ce schéma utilise également le modèle de l'oracle aléatoire. Bien que la sécurité de ce modèle soit remise en cause dans un contexte pratique [CGH98] [GK03], le mécanisme de format PSS reste inattaqué.

Oracle et oracle aléatoire

Le concept d'**oracle** est introduit afin de décrire l'entité infaillible auquel l'attaquant peut s'adresser à volonté afin d'obtenir la réponse d'un algorithme supposé ne pas exister.

Un oracle est dit **aléatoire** si l'information que l'attaquant possède sur cet oracle est strictement limitée aux réponses qu'il obtient à ces occasions : chaque réponse de cet oracle n'apporte aucune information sur les réponses ultérieures.

Le **modèle de l'oracle aléatoire** est un concept utilisé pour prouver qu'un schéma de signature est sûr, en aboutissant à la conclusion que l'algorithme symbolisé par

l'oracle, ne peut exister. La fonction de hachage est assimilée à une fonction parfaitement aléatoire. Mathématiquement, on pose la définition de l'oracle aléatoire comme suit :

Définition 4.1.9 *Pour toute constante k, un oracle aléatoire est une fonction H sélectionnée uniformément dans l'ensemble $\mathcal{H}_k$ des fonctions de $\{0,1\}^*$ vers $\{0,1\}^k$.*

4.1.4 Attaques connues des schémas de type Rabin-Williams

Les méthodes de padding, en particulier les fonctions de hachage, appliquées dans les schémas de type Rabin ou RSA sont la cible des attaques pour ce type de schémas.

Inspirés par l'article de H.C. Desmedt et A.M. Odlyzko [DO86], JS. Coron, D. Naccache et J. Stern [CNS99] présentent en 1999 une attaque, dont la complexité dépend de la taille du haché du message noté $\mu(M)$: la taille du module ne rentre pas en compte. Le principe de cette attaque consiste à rechercher des relations multiplicatives en se basant sur la probabilité non négligeable de constater le caractère l-friable pour des entiers de faible taille, avec l raisonnablement petit. Ainsi, lorsque la longueur de $\mu(M)$ est inférieure à 80 bits (ISO/IEC 9796-2), il est possible de forger une signature RSA, ou d'extraire la factorisation du module dans le cas du schéma de Rabin-Williams, à l'aide d'une attaque à messages choisis. Le nombre de messages que l'attaquant demande à signer dépend du nombre de facteurs premiers de $\mu(M)$. Les auteurs précisent cependant qu'un padding de longueur 160 bits comme SHA-1, peut être considéré comme sûr.

L'attaque de M. Joye et JJ. Quisquater [QJ01] en 2001 reprend les étapes de cette attaque en passant d'une falsification sélective à une falsification universelle : tout message peut être signé, après une attaque à messages choisis. La seule mais notable différence avec précédemment, est la non restriction des facteurs de l'élément $\mu(M)$. Ainsi, une signature RSA peut être forgée et la factorisation des schémas de type Rabin peut être déduite. Une généralisation de cette attaque à tout exposant public de vérification peut être également envisagée.

Les modules considérés lors de ces attaques sont composés de deux facteurs distincts congrus à 3 modulo 4. Une légère modification pourrait être apportée à cette attaque pour des modules quelconques : l'équation $p_j^{4d} = p_j^2$ deviendrait $p_j^{2^{b+1}d} = p_j^{2^b}$ où b est le niveau du module.

On rappelle cependant que cette attaque ne permet pas de mettre en péril la sécurité de ces schémas en prenant certaines précautions sur la longueur du padding.

Initiée par W. De Jonge et D. Chaum [dJC85], étendue par M. Girault et J.F. Misarsky [GM97], puis J.S. Coron [Cor00], une série d'attaques s'applique sur les paddings de nature affine, combinant les 3 types de redondances : multiplicative, additive et modulaire.

4.2 Etude chronologique des schémas de type Rabin-Williams

Les schémas de sécurité suivants basés sur le schéma de type Rabin-Williams ont parfois été proposés comme schémas de chiffrement. Nous convertissons si besoin, ces schémas en schémas de signature.

Dans la suite, on présente des schémas à partir du message déjà formaté, afin de simplifier les notations : on considère donc m tel que $0 \leq m < n$. Les schémas de Rabin, de Rabin-Williams et de Kurosawa-Ogata sont décrits de façon similaire afin de mettre en avant leur structure que nous retrouvons dans le nouveau schéma.

Il y a plus de vingt ans, M.O. Rabin [Rab79] propose une fonction à clé publique : un célèbre schéma de signature est né.

1. Génération des clés :
   Soient p et q deux nombres premiers et $n = pq$.
   Soient la clé publique $pk : n$, et la clé privée $sk : p, q$.
2. Génération de la signature : $Sign(m, pk, sk)$
   - Calcul d'une racine carrée s de m telle que $s^2 = m \mod n$
3. Envoi de la signature (m, s)
4. Vérification de la signature : $Verif(s, m, pk)$
   - Calcul de $\tilde{m}$ tel que $\tilde{m} = s^2 \bmod n$
   - Si $\tilde{m} = m$, validation de la signature, et rejet sinon.

Le calcul de la racine carrée s'effectue à l'aide d'un algorithme probabiliste comme l'algorithme de Tonelli-Shanks. Aucune restriction sur les modules n'est faite.

Un an plus tard, H.C. Williams [Wil80] propose une nouvelle fonction à clé publique, inspirée du schéma de Rabin. La sécurité de ce schéma est également prouvée équivalente à la factorisation des grands nombres, mais contrairement au schéma de Rabin, ce schéma fait correspondre une signature à un message unique. De cette façon, l'ambiguïté lors de la vérification de la signature sans redondance est inexistante. La sécurité de ce schéma est soulignée encore aujourd'hui par l'article de D. Bernstein [Berre], comme un exemple de schéma de signature, peu reconnu et peu utilisé à tort.

1. Génération des clés :
   Soient deux nombres premiers $p = 3 \bmod 8$, $q = 7 \bmod 8$ et $n = pq$.
   Soient $pk : n$, et $sk : d = (n - p - q + 5)/8$ les clés publique et privée.
2. Génération de la signature : $Sign(m, pk, sk)$
   - Si $(m|n) = 1$, calcul de s tel que $s = m^d \bmod n$
   - Si $(m|n) = -1$, calcul de s tel que $s = (m/2)^d \bmod n$
3. Envoi de la signature (m, s)
4. Vérification de la signature : $Verif(s, m, pk)$
   - Calcul de $\tilde{m} = s^2 \bmod n$

– Détermination de $a_1, a_2 \in \{0,1\}$ par une recherche exhaustive en calculant $m' = (-1)^{a_1} 2^{a_2} \tilde{m} \bmod n$ et vérification de l'égalité $m = m'$.

Cette solution permet l'existence d'un exposant privé d pour le calcul de racines carrées : un algorithme déterministe est donc appliqué. Mais, contrairement au schéma de signature de Rabin, seulement les entiers de Williams peuvent être utilisés (le module est le produit de 2 nombres premiers p, q tels que $p, q = 3 \bmod 4$ et $p \neq q \bmod 8$).

En 1984, H.C. Williams [Wil84] généralise sa méthode à tout module, produit de deux premiers distincts. La clé publique est composée du module, d'un exposant public et de deux nombres publics, souvent petits. Cette modification implique la transmission de deux nouveaux paramètres (deux bits) avec la signature. Cette généralisation utilise les irrationnels quadratiques.

En 1985, H.C. Williams [[Wil85],Partie 1] présente un nouveau schéma pour un module composé de deux facteurs p, q tels que $p, q = 3 \bmod 4$ en utilisant un nombre noté g_2 tel que $(g_2|n) = -1$ (le premier schéma utilise le cas $g_2 = 2$).

1. Génération des clés :
   Soient deux nombres premiers $p, q = 3 \bmod 4$ et $n = pq$.
   Soient $pk : (n, g_2)$ où $(g_2|p)(g_2|q) = -1$, et $sk : d = (n - p - q + 5)/8$ les clés publique et privée.
2. Génération de la signature : $Sign(m, pk, sk)$
   – Si $(m|n) = 1$, calcul de s telle que $s = m^d \bmod n$
   – Si $(m|n) = -1$, calcul de s telle que $s = (m/g_2)^d \bmod n$
3. Envoi de la signature (m, s)
4. Vérification de la signature : $Verif(s, m, pk)$
   – Calcul de $\tilde{m} = s^2 \bmod n$
   – Détermination de $a_1, a_2 \in \{0,1\}$ par une recherche exhaustive en calculant $m' = (-1)^{a_1} g_2^{a_2} \tilde{m} \bmod n$ et vérification de l'égalité $m = m'$.

En 1988, K. Kurosawa, T. Itoh et M. Takeuchi [KIT88] proposent une fonction à clé publique utilisant un «reciprocal number», aussi sûre que la factorisation des grands nombres. Ce schéma conserve l'avantage de la première version du schéma de Williams en utilisant deux nombres premiers arbitraires p et q. La majeure contribution de cet article est l'utilisation d'un nombre noté g_1 qui satisfait les propriétés $(g_1|p) = (g_1|q) = -1$. Deux bits qui dépendent du message sont toujours joints à la signature. Une preuve de sa sécurité est également donnée : casser le schéma proposé revient à résoudre le problème de la factorisation des grands nombres.

En 1999, K. Kurosawa et W. Ogata [KO99][KOMM01] proposent un schéma de signature efficace. Tous les modules peuvent être considérés dans le schéma de base. Deux nombres publics g_2 et g_3 sont choisis durant la génération de clés. Ils vérifient $(g_2|p) = (g_3|q) = -1$ et $(g_2|q) = (g_3|p) = 1$. Un des avantages de cette solution est la non nécessaire transmission des paramètres liés au message avec un module quelconque. Ce dernier schéma est le plus proche du nouveau schéma présenté.

1. Généralisation des clés :
   Soit p et q deux nombres premiers et $n = pq$.
   Clé publique pk : (n, g_1, g_2) avec

$$(g_1|p) = (g_2|q) = 1 \qquad \text{et} \qquad (g_1|q) = (g_2|p) = -1$$

   Clé privée sk : (p, q)
2. Généralisation de la signature : $Sign(m, pk, sk)$
   - Si $(m|p) = (m|q) = 1$, calcul de s tel que $s^2 = m \bmod n$
   - Si $(m|p) = (m|q) = -1$, calcul de s tel que $s^2 = mg_1g_2 \bmod n$
   - Si $(m|p) = 1$ et $(m|q) = -1$, calcul de s tel que $s^2 = mg_1 \bmod n$
   - Si $(m|p) = -1$ et $(m|q) = 1$, calcul de s tel que $s^2 = mg_2 \bmod n$
3. Envoi de la signature (m, s)
4. Vérification de la signature : $Verif(s, m, pk)$
   - Calcul de $m' = s^2 \bmod n$
   - Calcul de $m' = m$, $m' = mg_1$, $m' = mg_2$ et $m' = mg_1g_2$
     Si l'une des égalités est respectée, validation de la signature, et rejet sinon.

Dans cette solution, il y a seulement 4 cas à étudier lors de la recherche exhaustive pour un module quelconque. Cependant, tout comme le premier schéma de Rabin, l'algorithme probabiliste utilisé pour la génération de la signature dans les deux derniers schémas proposés, possède une complexité en $O(|n|^4)$.

Le Tableau 4.2 résume les particularités de chacun de ces schémas : la colonne *Module* indique la nature du module considéré, la colonne *Exposant* indique s'il existe un exposant d secret, la colonne *Symbole* indique si les propriétés des nombres de base sont liées aux symboles de Legendre, et la colonne *Paramètres* si des paramètres liés au message sont transmis.

	Module			Exposant		Symbole		Paramètres	
	Nbres Williams	$p, q = 3 \bmod 4$	Nbres qcques	non	oui	non	oui	non	oui
[Rab79]			⋆	⋆		⋆		⋆	
[Wil80]	⋆				⋆		⋆	⋆	
[Wil84]			⋆	⋆		⋆			⋆
[Wil85]		⋆			⋆		⋆	⋆	
[KIT88]			⋆	⋆			⋆		⋆
[KO99]			⋆	⋆			⋆	⋆	
ici			⋆		⋆		⋆	⋆	

TAB. 4.2 – Récapitulatif des schémas de type Rabin-Williams

Durant cette période de recherche portée sur des modules composés de deux facteurs quelconques, des solutions ont été proposées afin de généraliser l'exposant public 2 à de plus larges exposants. Dès 1985, H.C. Williams [[Wil85],Partie 2] propose une solution en considérant les entiers d'Eisenstein pour le cas de l'exposant 3, et sur le même raisonnement, R. Scheidler et H.C. Williams [SW95] proposent en 1995 une généralisation du schéma de Rabin-Williams à de plus larges exposants premiers. Ces exposants seront repris au cours de la généralisation du nouveau schéma présenté dans ce document.

En 1992, J.H. Loxton, D.S.P. Khoo, G.J. Bird et J. Seberry [LKBS92] puis R. Scheidler [Sch98] proposent des solutions dans le cas d'exposant public étendu RSA de la forme $3e$ où e est un exposant RSA. Le premier cas considère deux facteurs premiers p et q tels que $p = 4 \bmod 9$ et $q = 7 \bmod 9$, et le deuxième cas généralise à deux facteurs congrus tous deux à 1 modulo 3. La preuve d'équivalence avec la factorisation est également présente. Le raisonnement sur les racines de l'unité du deuxième article se retrouve dans le nouveau schéma proposé dans ce document, ainsi que les deux nombres publics ajoutés à la clé publique et les deux paramètres joints à la signature. Cependant, ces deux schémas proposés, bien que mathématiquement intéressants, ne permettent pas une implémentation facile dans un contexte réel.

Notre nouveau schéma concilie les performances de l'algorithme déterministe de la signature de Rabin-Williams à l'aide d'un exposant privé fixé, avec la non restriction du choix de modules de la signature, et en utilisant les propriétés des symboles de Legendre de 2 nombres de base ajoutés à la clé publique.

La Figure 4.1 illustre le positionnement du schéma de signature que nous présentons.

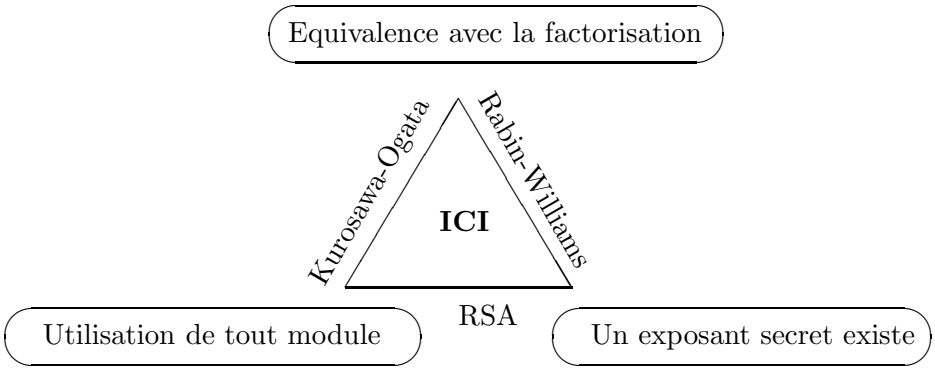


FIG. 4.1 – Problématique de la signature généralisée de Rabin-Williams

4.3 Spécifications de la généralisation de Rabin-Williams

Dans cette partie, nous présentons dans un premier temps le schéma de Rabin-Williams généralisé à tout module. Ensuite, nous justifions le choix des nombres de base

choisis dans les spécifications de ce nouveau schéma de signature à l'aide d'un théorème qui justifie les conditions suffisantes de ce choix. Avec une probabilité écrasante, il est possible de trouver ces deux nombres publics qui permettent d'obtenir un schéma avec une complexité en $O(|n|^3)$.

Nous utilisons les résultats obtenus au chapitre précédent pour le calcul des racines d'ordre impair, quel que soit le module considéré.

Théorème 4.3.1 *Soit $n = pq$, où p et q sont deux nombres premiers distincts tels que $p = 2^{b_p}p_1 + 1$ et $q = 2^{b_q}q_1 + 1$ avec b_p et b_q les niveaux de p et q.*
Soit V_n l'ensemble défini par la Proposition 1.2.7 et $b = max(b_p, b_q)$.
On pose

$$d = \frac{1}{2}\left(\frac{p-1}{2^{b_p}}\frac{q-1}{2^{b_q}} + 1\right) = \frac{1}{2}(p_1q_1 + 1)$$

Alors $f : x \mapsto x^d \bmod n$ et $g : x \mapsto x^2 \bmod n$, définies de l'ensemble V_n dans lui-même, sont des fonctions réciproques.

Preuve. cf Proposition 1.2.5

□

4.3.1 Schéma de Rabin-Williams généralisé à tout module

1. Génération des clés :

   On recherche deux nombres premiers p, q de taille suffisante pour que leur produit $n = pq$ ne puisse être factorisé avec les techniques de factorisation connues. On note $p = 2^{b_p}p_1 + 1$ et $q = 2^{b_q}q_1 + 1$ avec b_p et b_q les niveaux de p et q.
   On recherche deux petits entiers g_1, g_2 tels que :

   $$(g_1|p) = (g_1|q) = -1 \qquad \text{et} \qquad (g_2|n) = -1 \tag{4.1}$$

   Soient $pk = (n, g_1, g_2)$ et $sk = (p, q, d)$ les clés publique et privée. Le détenteur de la clé privée calcule d tel que

   $$d = \frac{1}{2}\left(\frac{p-1}{2^{b_p}}\frac{q-1}{2^{b_q}} + 1\right) = \frac{1}{2}(p_1q_1 + 1)$$

   Notons $b = max(b_p, b_q)$ et $b' = min(b_p, b_q)$, et soit $m \in \{0, ..., n-1\}$ le message à signer.
2. Génération de la signature $Sign(m, pk, sk)$:
   – Recherche de 2 entiers $a_1 \in \{0, 2^b - 1\}$ et $a_2 \in \{0, 2^{b'} - 1\}$ relatifs à m tels que :

   $$m^{p_1} = (g_1^{a_1}g_2^{a_2})^{p_1} \bmod p \qquad \text{et} \qquad m^{q_1} = (g_1^{a_1}g_2^{a_2})^{q_1} \bmod q$$

   – Calcul de $s = \left(mg_1^{-a_1}g_2^{-a_2}\right)^d \bmod n$
3. La signature $Sign(m, pk, sk)$ est le triplet (s, a_1, a_2)
4. Vérification de la signature $Verif(s, a_1, a_2, m, pk)$:
   – Calcul $m' = s^2g_1^{a_1}g_2^{a_2} \bmod n$
   – Si $m' = m$, validation de la signature, et rejet sinon.

La transmission des deux paramètres a_1, a_2 avec la signature est une possibilité mais non une nécessité : la connaissance de ces paramètres révèle seulement quelques bits d'information que quiconque peut déduire par une recherche exhaustive parmi les $2^{b_p+b_q}$ cas.

4.3.2 Théorème fondamental

Les conditions suffisantes sur les nombres de base de l'équation (4.1) vont être justifiées par le théorème suivant.

Le principe est la recherche de conditions sur les générateurs du 2-sous groupe de Sylow U_n : en divisant un message quelconque par une combinaison linéaire de ces générateurs, on obtiendra un élément de V_n pour lequel on pourra calculer une racine carrée, grâce au Théorème 4.3.1.

Théorème 4.3.2 *Soient p, q deux nombres premiers distincts impairs tels que $n = pq$. Soit U_n le 2-sous groupe de Sylow de $\mathbb{Z}_n^*$. Soient g_1, g_2 des entiers tels que*

$$(g_1|p) = (g_1|q) = -1 \qquad et \qquad (g_2|n) = -1$$

Posons $p - 1 = 2^{b_p} p_1$ et $q - 1 = 2^{b_q} q_1$ avec p_1, q_1 impairs. Alors $\eta_n(g_1)$ et $\eta_n(g_2)$ génèrent le groupe U_n, où η_n est définie par la Proposition 1.2.8.

Preuve. Posons $h_1 = \eta_n(g_1)$ et $h_2 = \eta_n(g_2)$. On rappelle que $\eta_p(g_1)$ et $\eta_q(g_1)$ appartiennent respectivement à U_p et à U_q, donc ce sont des éléments d'ordre 2. Alors, d'après la propriété des restes chinois, $\Xi(h_1) = \big(\eta_p(g_1), \eta_q(g_1)\big)$ aussi. Cela montre que h_1 est dans le 2-sous groupe de Sylow U_n de $\mathbb{Z}_n^*$. De la même façon, on obtient le même résultat pour h_2.

En utilisant la Proposition 1.2.9, on voit que $(h_1|p) = (g_1|p)^{p_1} = (-1)^{p_1} = -1$. Alors $h_1 \bmod p$ (qui appartient à U_p cf Proposition 1.2.8) n'est pas un carré. Alors son ordre est exactement 2^{b_p}. De la même façon, $h_1 \bmod q$ est dans U_q et d'ordre 2^{b_q}.

Comme $(g_2|p)$ et $(g_2|q)$ ont des valeurs opposées, supposons que $(g_2|p) = 1$ et $(g_2|q) = -1$ sans perte de généralité. De façon similaire, on a $h_2 \bmod q$ dans U_q et d'ordre 2^{b_q}. On a également $h_2 \bmod p$ dans U_p mais c'est un carré dans U_p et d'ordre **strictement moins** que 2^{b_p}, c'est à dire divisible par 2^{b_p-1}.

Alors on peut trouver k tel que $(h_1^2)^k = h_2$ modulo p. Posons $h'' = h_1^{-2k} h_2$. On a $h'' \bmod p = 1$. On a aussi $h'' \bmod q$ le produit dans U_q de $h_2 \bmod q$ (qui est d'ordre 2^{b_q}) et $h_1^{-2k} \bmod q$ (qui a un ordre divisible par 2^{b_q-1}). Alors $h'' \bmod q$ est d'ordre 2^{b_q} et génère U_q. Mais $h'' \bmod p = 1$ alors h'' est dans $\Xi_n^{-1}(\{1\} \times U_q)$ et est un générateur.

Comme $h_1 \bmod p$ est d'ordre 2^{b_p}, on peut d'abord trouver, pour chaque $u \in U_n$, un entier k_1 tel que $u = h_1^{k_1}$ modulo p. Alors, $uh_1^{-k_1} \in \Xi_n^{-1}(\{1\} \times U_q)$ et on peut trouver k'' tel que $uh_1^{-k_1} = h''^{k''}$. On obtient $u = h_1^{k_1} h''^{k''}$ et cela prouve que h_1 et h'' génèrent U_n. D'après l'expression de h'', on en déduit que h_1 et h_2 génèrent U_n.

□

Remarque : Soient $b = max(b_p, b_q)$ et $b' = min(b_p, b_q)$. D'après la preuve du théorème, on a que $\eta_n(g_1)$ est d'ordre 2^b modulo n. Comme le groupe U_n est d'ordre $2^{b+b'}$, on a :

$$U_n = \{\eta(g_1)^{a_1} \eta(g_2)^{a_2} | 0 \le a_1 < 2^b, 0 \le a_2 < 2^{b'}\}$$

4.3.3 Probabilité de succès pour la recherche de g_1 et g_2

Lors de la phase de génération de clés, on recherche deux petits nombres g_1 et g_2 qui satisfont les propriétés (4.1). Pour des raisons d'efficacité, on voudrait g_1, g_2 plus petits qu'une borne B. La proposition suivante montre que l'on peut trouver ces nombres avec une probabilité écrasante, même si l'on réduit la recherche à des nombres premiers.

Proposition 4.3.1 *Soit B un entier positif, et supposons $p, q \geq B$. Soit S l'ensemble des pairs (g_1, g_2) de premiers distincts, plus petits que B. Supposons que les valeurs $(x|p)$ et $(x|q)$ sont des variables indépendantes. Alors la proportion de pairs de l'ensemble S qui vérifie* (4.1) *tend asymptotiquement vers* $1/8$ *quand B tend vers $+\infty$.*

Preuve. La densité des nombres premiers g_1 tel que $(g_1|p) = -1$ est égal à $1/2$. La même affirmation s'applique à q : alors, la probabilité qu'un nombre g_1 donné, vérifie (4.1) est $1/4$. Pour le nombre g_2, il y a deux valeurs possibles pour $((g_2|p), (g_2|q))$ qui sont $(1, -1)$ et $(-1, 1)$. On en déduit une probabilité de $1/2$ pour g_2. On en déduit le résultat.

□

Par exemple, il y a 54 nombres premiers plus petits que 256. Il y a $54 \cdot 53 = 2862$ paires distinctes (g_1, g_2). Parmi eux, on a $54 \cdot 53/8 \simeq 358$ paires qui satisfont (4.1).

L'hypothèse de Riemann nous garantit l'existence de petits nombres «faux-carrés ». Ainsi, pendant la phase de génération, on recherche deux nombres publics parmi les premiers nombres premiers représentés sur un octet : deux petits nombres premiers plus petits que 255.

4.3.4 Complexité du schéma

Pour la complexité du protocole, la recherche des paramètres a_1, a_2 peut être négligée : c'est une recherche directe dans une table préalablement construite. La probabilité que le nombre de lignes et de colonnes soit petit est importante (cf Proposition 3.2.5). De même, la division et la multiplication modulaires du terme $g_1^{a_1} g_2^{a_2}$ peuvent être négligées : les probabilités d'avoir deux nombres de base g_1, g_2 représentés sur un octet seulement (cf Proposition 4.3.1), et d'avoir b_p et b_q petits, sont importantes.

Théorème 4.3.3 *La complexité du protocole généralisé de Rabin-Williams est en $O(|n|^3)$ pour la phase de génération de la signature, et en $O(|n|^2)$, pour la phase de vérification.*

Preuve. Il y a 2 exponentiations modulaires à effectuer du côté du signataire :
– recherche de a_1 et a_2
– utilisation de l'exposant privé d
Le vérifieur n'effectue, lui, qu'un carré modulaire.

□

4.3.5 Sécurité du nouveau schéma

Ce paragraphe décrit la preuve de sécurité du protocole généralisé de Rabin-Williams avec la factorisation des grands nombres.

Théorème 4.3.4 *Soient p, q deux nombres premiers et n tels que $n = pq$.*

Le schéma de signature de Rabin-Williams généralisé à tout module est sûr sous l'hypothèse que le problème de la factorisation du module est difficile.

Preuve.

Soit $b = max(b_p, b_q)$. On suppose que $b_p \leq b_q$ sans perte de généralité.

Si $b_p < b_q$ alors un vérifieur malhonnête choisit aléatoirement $m' \in \mathbb{Z}_n^*$. Dans la moitié des cas m' vérifie $(m'|q) = -1$ et $m'^{2^{b-1}} \notin V_N$. Il demande à l'oracle de signer m'^{2^b}. L'oracle retourne la signature s qui vérifie $s \in V_n$ et donc $s \neq \pm m'^{2^{b-1}}$. Par conséquence, le vérifieur malhonnête peut déduire la factorisation du nombre entier n en calculant $pgcd(s \pm m'^{2^{b-1}}, n)$.

Si $b_p = b_q$ alors le vérifieur malhonnête choisit aléatoirement $m' \in \mathbb{Z}_n^*$. Dans la moitié des cas, m' vérifie $(m'|p) \neq (m'|q)$, c'est à dire $m'^{2^{b-1}} \notin V_n$. Puis même raisonnement que précédemment.

□

Cette preuve est construite de la même façon que la preuve de sécurité du schéma de Rabin. Comme tout schéma de signature dont la sécurité est équivalente à la factorisation des grands nombres, il est vulnérable aux attaques à messages choisis, d'où la nécessité d'utiliser un mécanisme de format prouvé sûr. On se ramène au problème fondamental de la factorisation des grands nombres. Il n'est pas nécessaire de se baser sur le principe de l'oracle aléatoire pour démontrer la robustesse de ce schéma.

4.4 Généralisation du schéma avec l'exposant e

Bien que la complexité du schéma soit évidemment moins intéressante que dans le cas 2, il est intéressant de montrer, de la même façon que pour la généralisation du protocole GQ2 à un exposant public égal à $e = 2, 3, 4, 5, 7, 11$, que l'adaptation du schéma se fait aisément. Nous utilisons le théorème de Lenstra (cf Théorème 1.2.10).

Dans un deuxième temps, un mode de représentation des groupes multiplicatifs selon cet exposant public est présenté, afin de permettre de visualiser graphiquement le graphe $\{(x, y) \in \mathbb{Z}_p^*/\ x^e = y \mod p\}$ engendré, en particulier l'arbre unité du groupe. Trois exemples du protocole généralisé de Rabin-Williams à un exposant e sont décrits.

Théorème 4.4.1 *Soit $n = pq$, où p et q sont deux nombres premiers distincts tels que $p = e^{b_p} p_1 + 1$ et $q = e^{b_q} q_1 + 1$ avec b_p et b_q les niveaux de p et q.*

Soit V_n l'ensemble défini de la même façon que la Proposition 1.2.7 pour l'exposant e, et $b = max(b_p, b_q)$.

On pose

$$d = \frac{1}{e}\left(\frac{p-1}{e^{b_p}}\frac{q-1}{e^{b_q}} + 1\right) = \frac{1}{e}(p_1 q_1 + 1) \bmod p_1 q_1$$

Alors $f : x \mapsto x^d \bmod n$ et $g : x \mapsto x^e \bmod n$, définies de l'ensemble V_n dans lui-même, sont des fonctions réciproques.

Preuve. Existence de d : e est premier avec p_1 et q_1, donc avec p_1q_1.
e est donc inversible dans le groupe d'ordre p_1q_1.

Soit $a \in V_n$. Montrons que $a^{ed} = a \bmod n$.

$$a^{ed} = a^{p_1q_1+1} = a \bmod n$$

car $a^{p_1q_1+1} = 1 \bmod n$

□

4.4.1 Spécifications du schéma

Pour la généralisation de ce schéma, les facteurs doivent être des nombres premiers de 1^{ere} espèce pour la puissance e. Ainsi, pour le cas $e = 3$ par exemple, comme dans l'article de R. Scheidler [Sch98], les deux facteurs doivent être congrus à 1 modulo 3 : la moitié des nombres premiers ne peut donc être considérée.

1. Génération des clés :

   On recherche deux nombres premiers p, q de taille suffisante pour que leur produit $n = pq$ ne puisse être factorisé avec les techniques de factorisation connues. On note $p = e^{b_p}p_1 + 1$ et $q = e^{b_q}q_1 + 1$ avec b_p et b_q les niveaux de p et q.
   On recherche deux petits entiers g_1, g_2 tels que :

   $$(g_1|p)_e \neq 1 \qquad \text{et} \qquad (g_1|q)_e \neq 1 \qquad \text{et} \qquad (g_2|p)_e \neq 1 \qquad \text{et} \qquad (g_2|q)_e = 1 \tag{4.2}$$

   Soient $pk = (n, g_1, g_2)$ et $sk = (p, q, d)$ les clés publique et privée. Le détenteur de la clé privée calcule d tel que

   $$d = \frac{1}{e}\left(\frac{p-1}{e^{b_p}}\frac{q-1}{e^{b_q}} + 1\right) = \frac{1}{e}(p_1q_1 + 1) \bmod p_1q_1.$$

   Notons $b = max(b_p, b_q)$ et $b' = min(b_p, b_q)$, et soit $m \in \{0, ..., n-1\}$ le message à signer.

2. Génération de la signature : $Sign(m, pk, sk)$
   - Recherche de 2 entiers $a_1 \in \{0, e^b - 1\}$ et $a_2 \in \{0, e^{b'} - 1\}$ relatifs à m tels que :

     $$m^{p_1} = (g_1^{a_1}g_2^{a_2})^{p_1} \bmod p \qquad \text{et} \qquad m^{q_1} = (g_1^{a_1}g_2^{a_2})^{q_1} \bmod q$$

   - Calcul de $s = \left(mg_1^{-a_1}g_2^{-a_2}\right)^d \bmod n$
3. La signature $Sign(m, pk, sk)$ est le triplet (s, a_1, a_2)
4. Vérification de la signature : $Verif(s, a_1, a_2, m, pk)$:
   - Calcul $m' = s^e g_1^{a_1} g_2^{a_2} \bmod n$
   - Si $m' = m$, validation de la signature, et rejet sinon.

Théorème 4.4.2 *La complexité du protocole généralisé de Rabin-Williams est en $O(|n|^3)$ pour la phase de génération de la signature, et en $O(|n|^3)$, pour la phase de vérification.*

Bien que généralisable à d'autres exposants et intéressant mathématiquement, ce schéma possède une complexité qui augmente avec la taille de l'exposant public : l'exposant 2 est comme souvent le choix le plus intéressant.

Il est à noter également la restriction introduite sur les facteurs (deux facteurs congrus à 1 modulo 3 dans le cas de l'exposant 3), comme les articles sur la généralisation aux racines cubiques le présentaient.

4.4.2 Représentation des nombres premiers rationnels

Il est intéressant de représenter le graphe $\{(x, y) \in \mathbb{Z}_p^*/x^2 = y \bmod p\}$ pour visualiser la structure générale des groupes multiplicatifs $\mathbb{Z}_p^*$ où p premier impair de la forme $p = 2^{b_p}p_1 + 1$ où p_1 est impair. En effet, la notion de niveau des éléments est visualisée par le niveau des éléments dans le schéma. La généralisation de cette structure à de plus larges exposants $e = 2, 3, 4, 5, 7, 11$ est décrite, en reprenant les notations introduites dans le Chapitre 3.

Nous commençons par nous intéresser au «coeur » du groupe multiplicatif, l'arbre unité, à partir duquel la construction du groupe multiplicatif s'opère. L'algorithme de Toneli-Shanks permet l'élaboration de cet arbre unité.

Arbre unité

Proposition 4.4.1 *Soit G un groupe d'ordre 2^b. Il existe une suite décroissante de sous-groupes de G tel que :*

$\{1\} = G_b \subseteq G_{b-1} \subseteq ... \subseteq G_0 = G$ *tel que pour tout* $i \in \{0, ..., b\}$, $|G_i| = 2^{b-i}$

Définition 4.4.1 *Soit p un nombre premier.*

On appelle ***arbre unité*** *du groupe multiplicatif $\mathbb{Z}_p^*$, la représentation graphique du 2-sous groupe de Sylow de $\mathbb{Z}_p^*$ liée au graphe $\{(x, y) \in \mathbb{Z}_p^*/x^2 = y \bmod p\}$*

L'algorithme probabiliste à temps polynomial de **Tonelli-Shanks** permet de calculer une racine carrée de n'importe quel élément d'un groupe multiplicatif $\mathbb{Z}_p^*$: il utilise les b_p sous-groupes d'ordre $2^{b_p-1}, 2^{b_p-2}, ..., 1$ du 2-sous-groupe de Sylow de $\mathbb{Z}_p^*$, où b_p est le niveau de p

Les Figures 4.2 et 4.3 donnent des représentations graphiques de l'arbre unité avec η et i tels que $i^2 = -1 \bmod p$ et $\eta^2 = i \bmod p$.

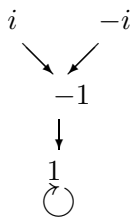


FIG. 4.2 – Schéma de l'arbre unité pour $e = 2$ et $b_p = 2$

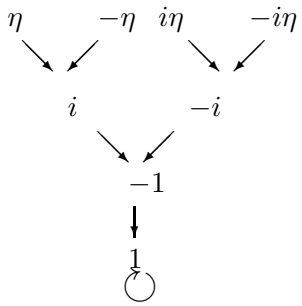


FIG. 4.3 – Schéma de l'arbre unité pour $e = 2$ et $b_p = 3$

Structure générale

Proposition 4.4.2 *Soit e un entier égal à 2, 3, 4, 5, 7, 11.*

Pour tout nombre premier p de 1^{ere} espèce pour la puissance e, on a l'isomorphisme suivant :

$$\mathbb{Z}_p^* \approx \mathbb{Z}/(p-1)\mathbb{Z} \approx \mathbb{Z}/e^{b_p}\mathbb{Z} \times \mathbb{Z}/p_1\mathbb{Z} \text{ avec } pgcd(e, p_1) = 1$$

où

- b_p *est le niveau de p*
- p_1 *est le nombre d'éléments d'ordre premier avec e.*

Théorème 4.4.3 *La probabilité de trouver un nombre premier de niveau b_p parmi l'ensemble des nombres premiers de 1^{ere} espèce pour la puissance e est de $1/e^{b_p}$.*

Théorème 4.4.4 *Tout nombre premier p de 1^{ere} espèce pour la puissance e est de niveau b_p si et seulement si $p = (e^{b_p}.i + 1) \bmod e^{i+1}$ avec $i \in \{1, ..., e-1\}$.*

Les Figures 4.4 et 4.5 illustrent la proportion de ces familles de facteurs. On remarque que 50% des nombres premiers sont congrus à 3 modulo 4.

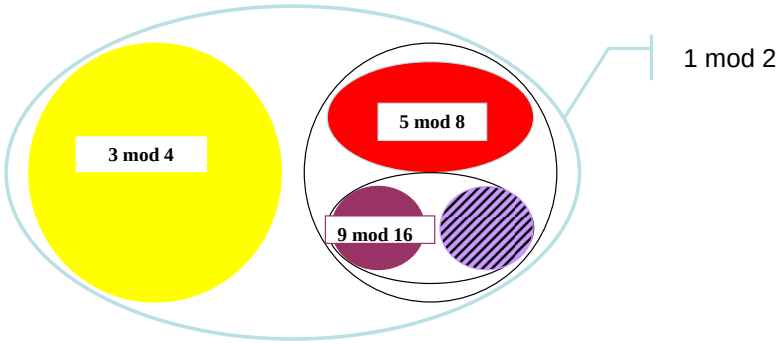


FIG. 4.4 – Classement des nombres premiers pour $e = 2$

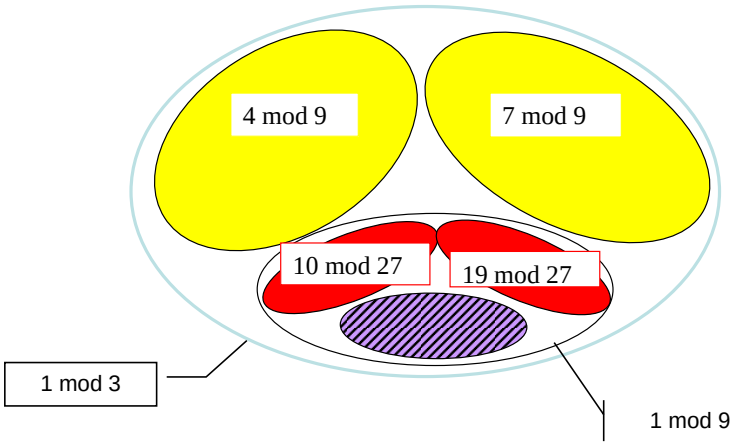


FIG. 4.5 – Classement des nombres premiers pour $e = 3$

Théorème 4.4.5 *Soit p un nombre premier de 1^{ere} espèce pour la puissance e.*
Soit $a \in V_p$ dans le groupe $\mathbb{Z}_p^*$ où $p = e^{b_p} p_1 + 1$.
La racine e^{ieme} $x \in V_p$ de a se calcule par :

$$x = a^y \bmod p \text{ avec } y = (p + (e^{b_p} - 1))/e^{b_p+1} = (p_1 + 1)/e \bmod p_1$$

Preuve. Analogue à celle de la Proposition 3.2.1.

□

Armature des groupes multiplicatifs

Définition 4.4.2 *Soit p un nombre premier de 1^{ere} espèce pour la puissance e.*
*On appelle **cycles** du groupe multiplicatif $\mathbb{Z}_p^*$, la représentation graphique du sous-groupe cyclique V_p liée au graphe $\{(x, y) \in \mathbb{Z}_p^* / x^e = y \bmod p\}$.*

Rappel : Chaque élément du groupe multiplicatif $\mathbb{Z}_p^*$ se décompose de manière unique sous la forme d'un élément de l'arbre unité et d'un élément appartenant aux cycles (cf Proposition 4.4.2). Un exemple d'illustration se trouve sur la Figure 4.6.

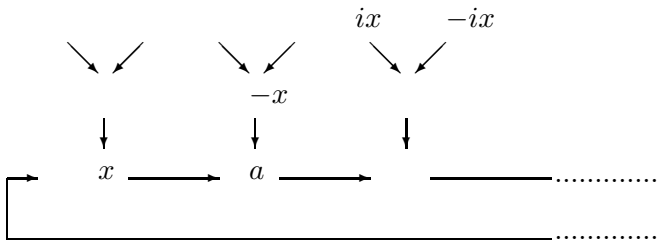


FIG. 4.6 – Représentation graphique pour $e = 2$

Construisons A un tableau d'entiers dont le i^{eme} élément correspond au nombre de cycles constitués de i éléments de V_p.

Entrée : p un nombre premier de 1^{ere} espèce pour la puissance e.
Sortie : A

$N \leftarrow p-1$
$i \leftarrow 1$
Tant que $N \neq 0$ **faire**
  $A(i) \leftarrow pgcd(e^i - 1, p-1)$
  Pour j **de** 1 **à** $i-1$ **faire**
    Si $i \bmod j = 0$ alors $A(i) \leftarrow A(i) - A(j)$ fin si
  fin pour
  $N \leftarrow N - A(i) \times e^{b_p}$
  $i \leftarrow i+1$
fin tant que
retourner(A)

Les arbres unités pour $e = 3$ sont élaborés à partir de l'algorithme modifié de Tonelli-Shanks. Graphiquement, la représentation est celle de la Figure 4.7 où $\omega^3 = 1 \bmod p$ et $\eta^3 = \omega \bmod p$.

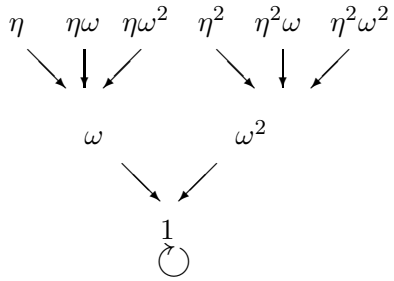


FIG. 4.7 – Schéma de l'arbre unité pour $e = 3$ et $b_p = 2$

Pour calculer des racines biquadratiques d'un élément a du groupe $\mathbb{Z}_p^*$, il suffit d'appliquer deux fois l'algorithme de Tonelli-Shanks.

On pose $\eta^4 = \gamma \bmod p$ et $\gamma^4 = 1 \bmod p$. (cf Figure 4.8)

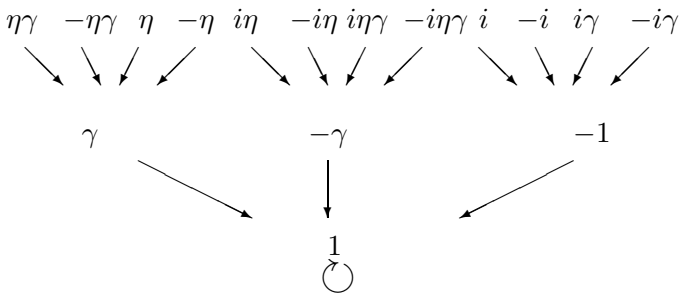


FIG. 4.8 – Schéma de l'arbre unité pour $e = 4$ et $b_p = 2$

Algorithme de Tonelli-Shanks pour les racines généralisées à la puissance e

Nous nous basons sur l'algorithme de Tonelli-Shanks [Coh93] qui calcule des racines carrées, pour élaborer l'algorithme généralisé à la puissance e.

But : recherche d'une racine e^{ieme} x d'un nombre a modulo p, où p est un nombre premier tel que $p = e^b p_1 + 1$ avec $pgcd(p_1, e) = 1$.

Entrée : a,p
Sortie : x

1. Choisir $0 < n < p$ aléatoirement jusqu'à ce que $(n|p)_e \neq 1$.
2. $m \leftarrow 1$
3. $x \leftarrow Recherche(b, p_1, m, p, a, n)$
4. Tant que $x = 0$ faire
   - $m \leftarrow m + 1$
   - $x \leftarrow Recherche(b, p_1, m, p, a, n)$
5. Retourner(x)

Algorithme *Recherche* :

Entrée : b, p_1, exp, p, a, n
Sortie : 0 ou x ou «a n'est pas un résidu pour la puissance e »

1. $z = (n^{exp})^{p_1} \bmod p$
2. $y \leftarrow z$
   $r \leftarrow b$
   $x \leftarrow a^{(p_1-(e-1))/e \bmod p_1} \bmod p$
   $E \leftarrow a^{(e-1)} x^e \bmod p$
   $x \leftarrow ax \bmod p$
3. Si $E = 1 \bmod p$, alors

si x est une racine e^{ieme} de a modulo p alors on retourne x sinon on retourne 0 fin si.

Sinon, on cherche le plus petit $m \geq 1$ tel que $E^{e^m} \neq 1 \bmod p$: si $m = r$ alors on retourne *a n'est pas un résidu pour la puissance e.*

4. $t \leftarrow y^{e^{r-m-1}} \bmod p$
   $y \leftarrow t^e \bmod p$
   $r \leftarrow m$
   $x \leftarrow xt \bmod p$
   $E \leftarrow Ey \bmod p$
   Test : si $r \neq 0$ retourner à l'étape 3 sinon retourner x

Au début de l'étape 3, on a $aE = x^e \bmod p$. On fait tourner l'algorithme jusqu'à ce que $E = 1 \bmod p$ pour obtenir x tel que $a = x^e \bmod p$. Pour cela, on parcourt les sous-groupes du e-sous groupe de Sylow : y est un générateur de ces sous-groupes successifs.

4.4.3 Exemples de la signature de Rabin-Williams généralisée

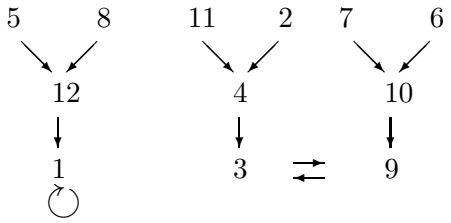
On détaille ici un exemple du schéma de signature de Rabin-Williams en illustrant une vision graphique des 4 étapes : le choix des 2 nombres publics, la génération de la table secrète, la génération de la signature et l'étape de vérification.

Exemple de la généralisation de Rabin-Williams pour $e = 2$

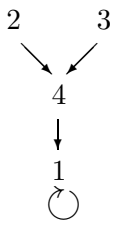
Représentation graphique

Soient deux nombres premiers $p = 13$ et $q = 5$. On a les écritures uniques :
$p = 2^2.3 + 1$ et $q = 2^2.1 + 1$. Voici la représentation graphique des 2 groupes correspondants où «$7 \rightarrow 10$ » signifie «$7^2 \bmod 13 = 10$ » :

$\mathbb{Z}_{13}^*$:

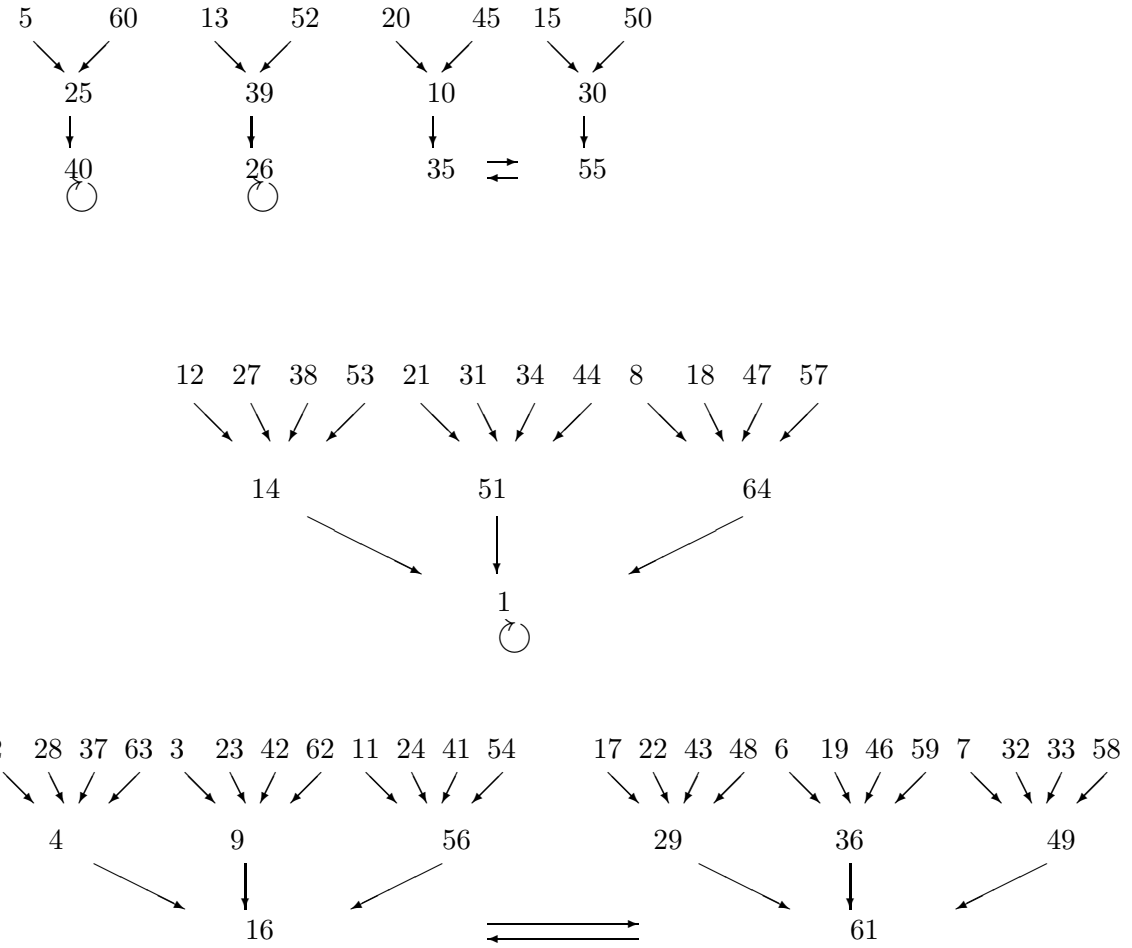


$\mathbb{Z}_5^*$:



Voici la représentation graphique du groupe $\mathbb{Z}_{65}^*$ relatif au module 65 :

$\mathbb{Z}_{65}^*$:



Choix des deux nombres publics

On remarque que les deux nombres de base $(2, 3)$ vérifient les hypothèses du théorème :

$$(2|p) = -1 \quad (2|q) = -1 \quad (3|p) = 1 \quad et \quad (3|q) = -1$$

Génération de la table secrète

Soit un tableau à deux dimensions qui dépend des 2 paramètres $a_1 \in \{0, 2^{b_p} - 1\}$ et $a_2 \in \{0, 2^{b_q} - 1\}$.

Chaque élément de ce tableau est le résultat du calcul de $\eta_{65}(2^{a_1}3^{a_2})$.

$a_1 \backslash a_2$	0	1	2	3
0	$\eta_{65}(1) = CRT(1,1) = 1$	$\eta_{65}(3) = CRT(1,3) = 53$	$\eta_{65}(9) = CRT(1,4) = 14$	$\eta_{65}(27) = CRT(1,2) = 27$
1	$\eta_{65}(2) = CRT(8,2) = 47$	$\eta_{65}(6) = CRT(8,1) = 21$	$\eta_{65}(18) = CRT(8,3) = 8$	$\eta_{65}(54) = CRT(8,4) = 34$
2	$\eta_{65}(4) = CRT(12,4) = 64$	$\eta_{65}(12) = CRT(12,2) = 12$	$\eta_{65}(36) = CRT(12,1) = 51$	$\eta_{65}(68) = CRT(12,3) = 38$
3	$\eta_{65}(8) = CRT(5,3) = 18$	$\eta_{65}(24) = CRT(5,4) = 44$	$\eta_{65}(72) = CRT(5,2) = 57$	$\eta_{65}(216) = CRT(5,1) = 31$

On remarque que l'on retrouve les éléments du 2-sous-groupe de Sylow de $\mathbb{Z}_{65}^*$.

Génération de la signature

Soit M un message à signer.

On n'utilisera pas PSS pour décrire cet exemple pour plus de clarté : $m = PSS(M)$.

On recherche les deux paramètres correspondants a_1 et a_2 de la valeur $\eta_{65}(m)$. (on sait que la valeur est nécessairement dans la table)

Par exemple, si $m = 56$:

On calcule $\eta_{65}(m) = CRT(12,1) = 51$.

En observant le tableau précédent, on peut en déduire que les 2 paramètres a_1 et a_2 sont égaux à 2.

Dans ce cas, le signataire calcule $s = (m/(2^2 3^2))^d = 16^d = 61 \bmod 65$ et envoie (m, s, a_1, a_2) au vérifieur.

Vérification

Le vérifieur calcule $m' = s^2 g_1^{a_1} g_2^{a_2} \bmod n$ et vérifie que $m = m'$.

Exemples de la généralisation de Rabin-Williams pour $e = 3$

Exemple 1

Soient 2 nombres premiers $p = 7$ et $q = 19$. Des exemples de représentation sont donnés dans l'annexe A.1. Voici la représentation graphique des 2 groupes correspondants où «$39 \rightarrow 1$ » signifie «$39^3 \bmod 133 = 1$ » :

On pose $n = pq$.

$$(2|p)_3 \neq 1 \quad (2|q)_3 \neq 1 \quad (11|p)_3 \neq 1 \quad et \quad (11|q)_3 = 1$$

Soit un tableau à deux dimensions qui dépend des 2 paramètres $a_1 \in \{0, 3^{b_q} - 1\}$ et $a_2 \in \{0, 3^{b_p} - 1\}$, où $b_p = 1$ et $b_q = 2$.

Chaque élément de ce tableau est le résultat du calcul de $\eta_{133}(2^{a_1} 11^{a_2})$.

$a_1 \backslash a_2$	0	1	2
0	1	121	11
1	4	85	44
2	16	74	43
3	64	30	39
4	123	120	23
5	93	81	92
6	106	58	102
7	25	99	9
8	100	130	36

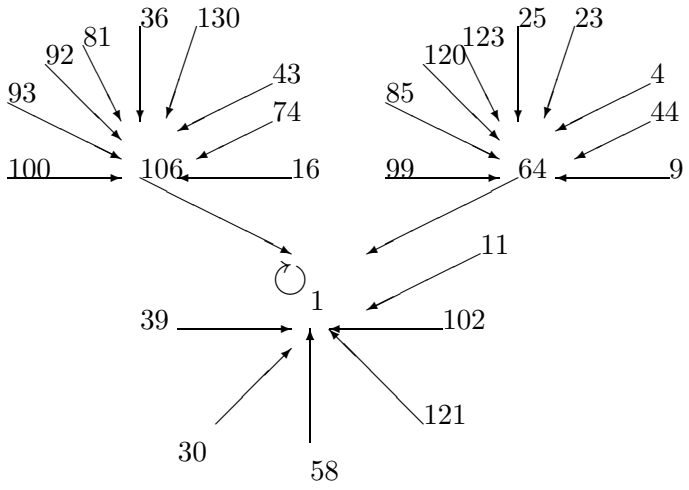
Génération de la signature

Soit m un message tel que $m = 99$:

On calcule $\eta_{133}(m) = 92$ et $d = 3$.

En observant le tableau précédent, on peut en déduire que $a_1 = 5$ et $a_2 = 2$.

Dans ce cas, le signataire calcule $s = (m/(2^5 11^2))^d = 113^d = 113 \bmod 133$ et envoie (m, s, a_1, a_2) au vérifieur.



Vérification

Le vérifieur calcule $m' = s^3 g_1^{a_1} g_2^{a_2} \bmod n$ et vérifie que $m = m'$.

Exemple 2

Soient 2 nombres premiers $p = 7$ et $q = 31$. Des exemples de représentation sont donnés dans l'annexe A.1.

On pose $n = pq$.

$$(2|p)_3 \neq 1 \quad (2|q)_3 \neq 1 \quad (5|p)_3 \neq 1 \quad (5|q)_3 = 1$$

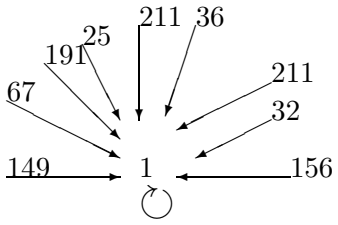
Soit un tableau à deux dimensions qui dépend des 2 paramètres $a_1 \in \{0, 3^{b_p} - 1\}$ et $a_2 \in \{0, 3^{b_q} - 1\}$, où $b_p = 1$ et $b_q = 1$.

Chaque élément de ce tableau est le résultat du calcul de $\eta_{217}(2^{a_1}5^{a_2})$.

$a_1 \backslash a_2$	0	1	2
0	1	67	149
1	32	191	211
2	156	36	25

Génération de la signature

Soit m un message tel que $m = 79$:
On calcule $\eta_{217}(m) = 25$ et $d = 7$.
En observant le tableau précédent, on peut en déduire que $a_1 = 2$ et $a_2 = 2$.
Dans ce cas, le signataire calcule $s = (m/(2^2 5^2))^d = 29^d = 120 \bmod 217$ et envoie (m, s, a_1, a_2) au vérifieur.



Vérification

Le vérifieur calcule $m' = s^3 g_1^{a_1} g_2^{a_2} \bmod n$ et vérifie que $m = m'$.

Chapitre 5

Preuve de sécurité non-interactive pour 2/3 des modules composés de 2 facteurs

Le «module » en cryptographie est l'entier composé sur lequel repose les protocoles basés sur la factorisation des grands nombres. Son caractère non factorisable avec les techniques actuellement connues, fonde la sécurité de ces schémas. Nous regardons ici comment maîtriser sa composition en facteurs premiers, sans connaître sa factorisation. Ce chapitre est issu d'un travail mené personnellement.

La validité du contenu de la clé publique est assurée en général par l'autorité de confiance qui décerne le certificat de cette clé. Dans le schéma de Rabin-Williams, le module distribué doit être un entier de Williams, et tout vérifieur doit pouvoir vérifier la nature de ce module par lui-même s'il ne peut l'obtenir de l'autorité de confiance. Cette preuve qui garantit le respect des spécifications du protocole, ne doit cependant pas révéler d'information sur les facteurs de ce module.

Dans un premier temps, nous présentons les articles sur lesquels nous nous sommes basés et qui abordent cette problématique de validité du module. En particulier, l'article de M. Gennaro, D. Micciancio et T. Rabin [GMR98] expose une série de preuves non-interactives zero-knowledge simples et efficaces, où l'ensemble des modules est regroupé par famille de modules.

Le but de ce chapitre est de mettre en évidence la possibilité d'extension d'une étape de cette série de preuves à de petits nombres premiers, en utilisant les propriétés de leurs symboles de Legendre. On propose également une combinaison de ces extensions permettant de construire un protocole qui couvre jusqu'à 2/3 des modules composés de deux facteurs, avec un apport limité d'information.

5.1 Etat de l'art et définitions

5.1.1 Etat de l'art

Pendant dix ans, les articles sur le problème de la validité du module ont construit des preuves pour les modules composés de safe primes ([CM99]), de quasi-safe primes ([GMR98]) ainsi que plus largement de deux facteurs quelconques ([GMR98]). Les deux premiers ensembles de modules sont considérés comme plus sûrs par les cryptologues, compte tenu de la construction particulière de leurs facteurs intégrant des grands nombres premiers ou des puissances de grands nombres premiers.

En 1987, J. Van de Graaf et R. Peralta [GP88] sont les premiers à introduire la problématique de la validité de la clé publique. Ils présentent le premier protocole sûr à divulgation nulle de connaissance capable de convaincre tout vérifieur que le module n est de la forme $n = p_1^{\alpha_1} p_2^{\alpha_2}$ où p_1 et p_2 sont congrus à 3 modulo 4, avec α_1 et α_2 impairs sous la forme introduite par Blum [Blu81].

En 1991, M. Blum, A. De Santis, S. Micali et G. Persiano [BSMP91] présentent des algorithmes sûrs et efficaces pour vérifier l'appartenance d'éléments à des langages NP-complets. Les preuves Zero-Knowledge Non-Interactives (NIZK) sont introduites.

En 1998, M. Gennaro, D. Micciancio et T. Rabin [GMR98] utilisent ce modèle, ainsi que le protocole zero-knowledge [BFL94] permettant de prouver qu'un nombre donné est square-free, pour construire une série de preuves indépendantes, simples et efficaces, statistiquement ou parfaitement zero-knowledge. Ce sont les notations et les preuves de cet article que nous reprenons dans ce chapitre.

En 1999, J. Camenish et M. Michels [CM99] complètent ces preuves en utilisant le test de primalité de Lehmann pour prouver la primalité des éléments $(p_1 - 1)/2$ et $(p_2 - 1)/2$. Ce rapport technique contient la preuve la plus complète, statistiquement sans apport d'information, pour les modules les plus utilisés dans le domaine de la cryptographie : les modules produits de deux safe primes.

Dans [GMR98], les modules considérés sont composés de facteurs dits quasi-safe primes. Mais quelle est la proportion que représente cette famille sur l'ensemble des modules composés de deux facteurs ? La série de preuves permet cependant de valider de manière générale les modules composés de deux facteurs, et nous ne prétendons pas améliorer ici ce résultat, mais seulement proposer une extension d'une étape de ces preuves afin de mettre en lumière la proportion des modules considérés dans cette étape : de 1/4 des modules (ceux composés de facteurs congrus 3 modulo 4, en particulier les safe primes) ou 3/8 des modules (en particulier les quasi-safe primes), on conclut à une couverture de 2/3 de ces modules par extension.

5.1.2 Définitions

Nous définissons le concept de preuves non-interactives zero-knowledge introduites par [BSMP91]. Ces preuves possèdent la particularité du partage d'une valeur aléatoire

R entre le prouveur et le vérifieur. Le livre de Oded Goldreich[Gol01] rappelle la théorie de ces preuves.

Nous reprenons, comme pour la généralisation du schéma de Rabin-Williams, les propriétés de résiduosité quadratique des petits nombres premiers par rapport à chacun des deux facteurs. Les symboles de Legendre des quatre premiers nombres premiers sont élaborés à partir des étapes décrites dans le livre de H. Cohen [Coh93].

Définition 5.1.1 *Une paire de machines (P, V) est appelée système de* ***preuve non-interactive*** *pour un langage L si V est à temps polynomial et que les deux conditions suivantes sont respectées : soit $\epsilon > 0$*

- *Complétude : Pour tout $x \in L$,*

$$Pr(V(x, R, P(x, R)) = 1) \geq 1 - \epsilon$$

où R est une variable aléatoire

- *Significativité : Pour chaque $x \notin L$ et pour tout algorithme B*

$$Pr(V(x, R, B(x, R)) = 1) \leq \epsilon$$

où R est une variable aléatoire.

La variable R est appelé la ***chaîne commune de référence****.*

Définition 5.1.2 *Soit n le produit de deux nombres premiers p, q. Soit $x \in \mathbb{Z}_n^*$. On définit la fonction σ_n par :*

$$\begin{array}{rccc} \sigma_n : & \mathbb{Z}_n^* & \longrightarrow & \{-1,1\}^2 \\ & x & \longmapsto & ((x|p),(x|q)) \end{array}$$

Définition 5.1.3 *Soit n le produit de deux nombres premiers p, q. Soit $x \in \mathbb{Z}_n^*$. On définit la fonction Q_n par :*

$$\begin{array}{rccl} Q_n : & \mathbb{Z}_n^* & \longrightarrow & \{-1,1\} \\ & x & \longmapsto & Q_n(x) = 1 \textit{ si } x \textit{ est un résidu quadratique} \\ & & & Q_n(x) = 0 \textit{ sinon} \end{array}$$

Proposition 5.1.1

$$(-1|p) = \begin{cases} 1 \textit{ si } p = \pm 1 \bmod 4 \\ -1 \textit{ si } p = \pm 3 \bmod 4 \end{cases} \quad (2|p) = \begin{cases} 1 \textit{ si } p = \pm 1 \bmod 8 \\ -1 \textit{ si } p = \pm 3 \bmod 8 \end{cases}$$

$$(3|p) = \begin{cases} 1 \textit{ si } p = \pm 1 \bmod 12 \\ -1 \textit{ si } p = \pm 5 \bmod 12 \end{cases} \quad (5|p) = \begin{cases} 1 \textit{ si } p = \pm 1 \bmod 5 \\ -1 \textit{ si } p = \pm 2 \bmod 5 \end{cases}$$

$$(7|p) = \begin{cases} 1 \textit{ si } p = \pm 1, \pm 3, \pm 9 \bmod 28 \\ -1 \textit{ si } p = \pm 5, \pm 7, \pm 11, \pm 13 \bmod 28 \end{cases}$$

Notation : On indice par la suite ces petits nombres premiers : $g_1 = 2$, $g_2 = 3$, $g_3 = 5$, $g_4 = 7$.

5.1.3 Familles de modules

Nous reprenons ici les notations proposées par [GMR98] afin de regrouper les modules par famille de modules.

- ODD : l'ensemble des entiers impairs

$$ODD = \{n \in \mathbb{N}/n = p_1^{\alpha_1}...p_f^{\alpha_f}, \alpha_1, ..., \alpha_f \in \mathbb{N}^*, p_1...p_f \in \mathbb{P}\}$$

- ODD' : l'ensemble des éléments de ODD possédant les conditions suivantes :

$$ODD' = \{n \in ODD/n = p^{\alpha_1}q^{\alpha_2}, \alpha_1, \alpha_2 = 1\}$$

- PPP (Prime Power Product) : l'ensemble des éléments de ODD possédant au plus deux facteurs premiers impairs :

$$PPP = \{n \in ODD/n = p^{\alpha_1}q^{\alpha_2}, \alpha_1, \alpha_2 \neq 0\}$$

- $PPP^{(1)}$: l'ensemble des éléments de PPP possédant deux conditions supplémentaires :

$$PPP^{(1)} = \{n \in PPP/n = p^{\alpha_1}q^{\alpha_2}, p, q \neq 1 \bmod 8 \ et \ p \neq q \bmod 8\}$$

- GP (les entiers de Van de Graaf et Perralta) : l'ensemble des éléments de PPP possédant deux conditions supplémentaires :

$$GP = \{n \in PPP/n = p^{\alpha_1}q^{\alpha_2}, p, q = 3 \bmod 4, \alpha_1, \alpha_2 \text{ impairs}\}$$

- W (les entiers de Williams) : l'ensemble des éléments de GP possédant les conditions supplémentaires :

$$W = \{n \in GP/n = p^{\alpha_1}q^{\alpha_2}, p \neq q \bmod 8, \alpha_1, \alpha_2 = 1\}$$

- B (les entiers de Blum) : l'ensemble des éléments de GP possédant la condition suivante :

$$B = \{n \in GP/n = p^{\alpha_1}q^{\alpha_2}, \alpha_1, \alpha_2 = 1\}$$

- $QSPP$: l'ensemble de ODD' tel que :

$$QSPP = \{n \in ODD'/n = PQ, P = 2p^{\alpha} + 1, Q = 2q^{\beta} + 1, p, q \text{ premiers}/ \\ P, Q, p, q \neq 1 \bmod 8, P \neq Q \bmod 8, p \neq q \bmod 8\}$$

- SF : l'ensemble des nombres impairs square-free :

$$SF = \{n \in ODD/\forall m > 1, m^2 \not| n\}$$

- $PP = SF \cap PPP$
- SF' : le sous-ensemble de SF tels que :

$$SF' = \{n \in SF/\forall p, q \ tels \ que \ p, q \ divise \ n, \ p \not| (q-1)\}$$

- DPP : l'ensemble de ODD' tels que :

$$DPP = \{n \in ODD'/n = pq \ \text{ avec } p, q \text{ distincts}\}$$

5.2 Séquence de preuves

Après l'introduction de notations et définitions supplémentaires, nous rappelons les trois étapes de la preuve décrite dans [GMR98]. La démonstration de la seconde étape sera détaillée : elle généralise le résultat introduit dans cet article avec les petits nombres premiers g_i.

5.2.1 Notations et définitions supplémentaires

On considère quatre ensembles A_i, B_i, C_i, D_i qui dépendent du i^{eme} nombre premier g_i et définis par :

- $A_i = \{p \in \mathbb{P} / p = 3 \bmod 4 \ et \ (g_i|p) = 1\}$
- $B_i = \{p \in \mathbb{P} / p = 3 \bmod 4 \ et \ (g_i|p) = -1\}$
- $C_i = \{p \in \mathbb{P} / p = 1 \bmod 4 \ et \ (g_i|p) = -1\}$
- $D_i = \{p \in \mathbb{P} / p = 1 \bmod 4 \ et \ (g_i|p) = 1\}$

D'après les résultats sur les symboles de Legendre (cf Proposition 5.1.1), on peut établir le Tableau 5.1. On note K_i le modulus nécessaire à la description de la répartition des nombres premiers selon le nombre g_i dans les ensembles A_i, B_i, C_i, D_i.

g_i	2	3	5	7
K_i	8	12	20	28
A_i	7 mod 8	11 mod 12	11, 19 mod 20	3, 19, 27 mod 28
B_i	3 mod 8	7 mod 12	3, 7 mod 20	7, 11, 15, 23 mod 28
C_i	5 mod 8	5 mod 12	13, 17 mod 20	5, 13, 17, 21 mod 28
D_i	1 mod 8	1 mod 12	1, 9 mod 20	1, 9, 25 mod 28

TAB. 5.1 – Classement des nombres premiers en fonction de petits nombres premiers

La Figure 5.1 schématise la répartition de ces nombres premiers. La Proposition 5.2.2 avec le petit nombre premier $g_1 = 2$, couvre les modules de l'ensemble $PPP^{(1)}$ composés de deux nombres premiers P, Q tel que $P, Q \notin D_1$ et $P \neq Q \bmod K_1$. Une preuve identique va être proposée ici pour $g_2, g_3 \ et \ g_4$.

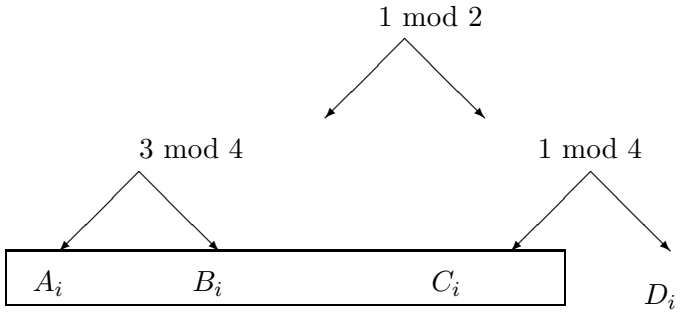


FIG. 5.1 – Graphe du classement des nombres premiers selon g_i

Définition 5.2.1 *On définit l'ensemble $PPP^{(i)}$ par l'ensemble des éléments de PPP tels que :*

$$PPP^{(i)} = \{n \in PPP / n = p^{\alpha_1} q^{\alpha_2}, p, q \neq 1 \bmod K_i \text{ et } p \neq q \bmod K_i\}$$

5.2.2 Etapes de la preuve complète

Dans l'article [GMR98], les auteurs décrivent les 3 étapes pour prouver que le module n est dans $ODD' \cap PPP^{(1)}$:

- $n \in SF'$
- $n \in PPP^{(1)}$
- $n \in DPP$

Les 3 propositions suivantes sont les 3 preuves NIZK correspondantes. La seconde étape est généralisée à tout petit nombre premier g_i pour obtenir une preuve zero-knowledge non-interactive pour l'ensemble $ODD' \cap PPP^{(i)}$.

Proposition 5.2.1 *Le protocole NIZK suivant permet de prouver qu'un nombre impair appartient à SF' :*

- *Entrée commune : n un nombre impair*
- *Entrée aléatoire : $x \in \mathbb{Z}_n^*$*
- *Prouveur :*
  - *Calcul de $M = n^{-1} \bmod \phi(n)$ et retourne $y = x^M \bmod n$*
  - *Envoie y*
- *Vérifieur : Accepte si $y^n = x \bmod n$*

est un protocole avec un seuil de sécurité de $1/d$ pour le langage SF', où d est le plus petit facteur de n.

Nous dirigeons le lecteur vers l'article [GMR98] qui développe la démonstration de cette proposition.

Proposition 5.2.2 *Soit g_i un petit nombre premier ($g_i = 2, 3, 5$ ou 7). Le système de preuve non-interactive pour l'ensemble des résidus quadratiques de $\mathbb{Z}_n^*$ tel que :*

- *Entrée commune : n un entier impair*
- *Entrée aléatoire : $x \in \mathbb{Z}_n^*$*
- *Prouveur : Calcul d'une racine carrée r modulo n d'un des éléments de l'ensemble $\{\pm x, \pm g_i x\}$*
- *Vérifieur : Accepte si r^2 est congru à $\pm x$ ou $\pm g_i x$ modulo n*

est un protocole zero-knowledge avec un seuil de sécurité de $1/2$ pour le langage $PPP^{(i)}$.

Preuve.

Montrons que la Proposition 5.2.2 est bien une preuve zero-knowledge pour le langage $PPP^{(i)}$: elle vérifie les propriétés de complétude, de significativité et de divulgation nulle de connaissance.

Complétude :

Soit g_i un petit nombre premier et l'ensemble $\{A_i, B_i, C_i, D_i\}$ relatif à ce nombre.

Soit $n \in PPP^{(i)}$. Cela signifie $n = P^\alpha Q^\beta$ pour P et Q tel que $P, Q \notin D_i$ et $P \neq Q \bmod K_i$.

– Cas 1 : $P \in B_i$ et $Q \in A_i$, $-1 = (-1,-1)$ et $g_i = (-1,1)$
Alors, pour tout $x \in \mathbb{Z}_n^*$, seulement un nombre $\pm x, \pm gx$ est un carré modulo $n = PQ$.
– Cas 2 : $P \in C_i$ et $Q \in A_i$, $-1 = (1,-1)$ et $g_i = (-1,1)$
Alors, pour tout $x \in \mathbb{Z}_n^*$, seulement un nombre $\pm x, \pm g_i x$ est un carré modulo $n = PQ$.
– Cas 3 : $P \in C_i$ et $Q \in B_i$, $-1 = (1,-1)$ et $g_i = (-1,-1)$
Alors, pour tout $x \in \mathbb{Z}_N^*$, seulement un nombre $\pm x, \pm g_i x$ est un carré modulo $n = PQ$.

Alors, le prouveur peut toujours calculer une racine carrée modulo n d'un, et seulement un, des éléments $\pm x, \pm g_i x$, et le vérifieur accepte toujours.

Complétude								
Module		$\sigma_n(-1)$	$\sigma_n(g)$	$\sigma_n(x)$	$Q_n(?) =$			
p	q				x	$-x$	gx	$-gx$
B_i	A_i	$(-1,-1)$	$(-1,1)$	$(1,1)$	1	0	0	0
				$(-1,1)$	0	0	1	0
				$(-1,-1)$	0	1	0	0
				$(1,-1)$	0	0	0	1
C_i	A_i	$(1,-1)$	$(-1,1)$	$(1,1)$	1	0	0	0
				$(-1,1)$	0	0	1	0
				$(-1,-1)$	0	0	0	1
				$(1,-1)$	0	1	0	0
C_i	B_i	$(1,-1)$	$(-1,-1)$	$(1,1)$	1	0	0	0
				$(-1,1)$	0	0	0	1
				$(-1,-1)$	0	0	1	0
				$(1,-1)$	0	1	0	0

Significativité :

Supposons que $n \notin PPP^{(i)}$. On prouve qu'aucun prouveur ne peut convaincre un vérifieur avec une probabilité supérieure à 1/2.

– Cas 1 : Si $n \notin PPP$ alors n est composé par 3 ou plus de trois facteurs premiers impairs.
La probabilité de trouver un nombre, aléatoirement, $x \in \mathbb{Z}_n^*$ tel que x est un résidu quadratique, est au plus $1/2^3 = 1/8$. Alors, la probabilité qu'aucun des 4 éléments $\pm x, \pm g_i x$ ait une racine carrée modulo n est supérieure à 1/2
Un vérifieur honnête rejette un fraudeur avec une probabilité supérieure à 1/2.
– Cas 2 :
Si $n \in PPP$ mais que $n \notin PPP^{(i)}$ alors n est composé par au plus 2 facteurs P et Q tels que $P \in D_i$, $Q \in D_i$ ou $P = Q \bmod K_i$

Il y a 1/4 de résidus quadratiques dans $\mathbb{Z}_n^*$, alors :
- Si $P, Q \in D_i$, la probabilité d'avoir un résidu quadratique dans l'ensemble $\{\pm x, \pm g_i x\}$ est au plus $\frac{4}{4} \times \frac{1}{4} + \frac{0}{4} \times \frac{3}{4} = \frac{1}{4}$
- Si $P = Q \bmod K_i$ mais $P, Q \notin D_i$, la probabilité d'avoir un résidu quadratique dans $\{\pm x, \pm g_i x\}$ est au plus $\frac{2}{4} \times \frac{1}{4} + \frac{2}{4} \times \frac{3}{4} = \frac{1}{2}$
- Si $P \in D_i$ mais $Q \notin D_i$ (sans restriction de généralité), la probabilité d'avoir un résidu quadratique dans $\{\pm x, \pm g_i x\}$ est au plus $\frac{2}{4} \times \frac{1}{4} + \frac{2}{4} \times \frac{3}{4} = \frac{1}{2}$

Alors, le vérifieur rejette le fraudeur avec une probabilité supérieure à 1/2.

Significativité								
Module		$\sigma_n(-1)$	$\sigma_n(g)$	$\sigma_n(x)$	$Q_n(?) =$			
p	q				x	$-x$	gx	$-gx$
A_i	A_i	$(-1,-1)$	$(1,1)$	$(1,1)$	1	0	1	0
				$(-1,1)$	0	0	0	0
				$(-1,-1)$	0	1	0	1
				$(1,-1)$	0	0	0	0
B_i	B_i	$(-1,-1)$	$(-1,-1)$	$(1,1)$	1	0	0	1
				$(-1,1)$	0	0	0	0
				$(-1,-1)$	0	1	1	0
				$(1,-1)$	0	0	0	0
C_i	C_i	$(1,1)$	$(-1,-1)$	$(1,1)$	1	1	0	0
				$(-1,1)$	0	0	0	0
				$(-1,-1)$	0	0	1	1
				$(1,-1)$	0	0	0	0
D_i	D_i	$(1,1)$	$(1,1)$	$(1,1)$	1	1	1	1
				$(-1,1)$	0	0	0	0
				$(-1,-1)$	0	0	0	0
				$(1,-1)$	0	0	0	0

D_i	A_i	$(1,-1)$	$(1,1)$	$(1,1)$	1	0	1	0
				$(-1,1)$	0	0	0	0
				$(-1,-1)$	0	0	0	0
				$(1,-1)$	0	1	0	1
D_i	B_i	$(1,-1)$	$(1,-1)$	$(1,1)$	1	0	0	1
				$(-1,1)$	0	0	0	0
				$(-1,-1)$	0	0	0	0
				$(1,-1)$	0	1	1	0
D_i	C_i	$(1,1)$	$(1,-1)$	$(1,1)$	1	1	0	0
				$(-1,1)$	0	0	0	0
				$(-1,-1)$	0	0	0	0
				$(1,-1)$	0	0	1	1

Zero-knowledge :

On peut en effet construire un simulateur, qui choisit aléatoirement deux nombres r et α tels que $r \in \mathbb{Z}_n^*$ et $\alpha \in \{\pm 1, \pm g_i\}$. Il calcule alors $x = r^2/\alpha \bmod n$ et renvoie (x, r) comme couple de solution valide.

□

Proposition 5.2.3 *Le protocole NIZK suivant permet de prouver qu'un nombre de l'ensemble PP appartient à DPP :*

- *Entrée commune : $n \in PP$*
- *Entrée aléatoire : $x \in \mathbb{Z}_n^*$*
- *Prouveur :*
  - *Calcule $x^M \bmod n$ où $M.\pi_1(n-1) = 1 \bmod \phi(n)$ ($\pi_1(n-1)$ le plus grand diviseur impair de $n-1$)*
  - *Envoie y*
- *Vérifieur : Accepte si n n'est pas un nombre de la forme $2^k + 1$ (nombre de Mersenne) et vérifie ensuite $y^{\pi_1(n-1)} = x \bmod n$*

est un protocole avec un seuil de sécurité de $1/d$ pour le langage DPP dans PP, où d est le plus petit facteur de $\pi_1(n-1)$.

Nous dirigeons le lecteur vers l'article [GMR98] qui développe la démonstration de cette proposition.

5.3 Combinaison des protocoles en un protocole unique

Par la proposition ci-dessous, on prouve la validité de la clé publique contenant un module de la forme $n = pq$ dans 2/3 des cas, en combinant les extensions proposées pour la deuxième étape de la séquence de preuves : on obtient une preuve d'appartenance du module à l'union des ensembles $PPP^{(i)}$ pour i variant de 1 à 4, noté P.

Proposition 5.3.1 *Le protocole suivant permet de prouver qu'un nombre impair appartient à* $P = \cup_{i=1}^{4} PPP^{(i)}$ *:*

- *Entrée commune : n un nombre impair*
- *Entrée aléatoire :* $x \in \mathbb{Z}_n^*$
- *Prouveur :*
  - *Etape 1 : Choisit un nombre premier g parmi l'ensemble* $\{2, 3, 5, 7\}$ *de manière aléatoire.*
  - *Etape 2 : Calcule une racine carrée r modulo n d'un élément de l'ensemble* $\{\pm x, \pm gx\}$*, s'il existe. S'il n'existe pas, retour à l'étape 1.*
  - *Etape 3 : Envoie* (r, g)
- *Vérifieur : Accepte si* r^2 *est congru à* $\pm x$ *ou* $\pm gx$ *modulo n*

est une preuve d'appartenance au langage P, avec un seuil de sécurité de 1/2.

L'apport d'information du nombre aléatoire g est à apport limité de connaissance, chaque g_i recouvrant 3/8 des modules de manière uniforme.

5.4 Evolution de 3/8 aux 2/3 des modules

Dans cette partie, on évalue la proportion de modules concernés par la nouvelle preuve non-interactive où l'on remplace la deuxième étape de l'article [GMR98] par la combinaison des quatre protocoles définis selon les premiers nombres premiers. Chacun de ces protocoles couvre 3/8 des entiers contenus dans l'ensemble ODD'. Leur combinaison élève cette proportion à près de 2/3 ; le nombre de petits nombres premiers considérés permet de se rapprocher de cette borne maximale.

5.4.1 Vers une probabilité de couverture de 3/8 des modules...

Pour tout $i \in \{1, 2, 3, 4\}$, on fait l'approximation suivante
$Pr(p \in A_i) \approx Pr(p \in B_i) \approx Pr(p \in C_i) \approx 1/4$, selon l'hypothèse que les nombres premiers sont répartis uniformément (cf Graphe 5.1).

La proposition suivante démontre que la preuve NIZK pour le langage $PPP^{(i)}$ recouvre une proportion de module de l'ensemble ODD' au moins égale à 3/8.

Proposition 5.4.1 *Pour tout* $i \in \{1, 2, 3, 4\}$*, on a*

$$Pr(n \in ODD' \cap PPP^{(i)}) \approx 2(Pr(p \in A_i)Pr(q \in B_i)) + 2(Pr(p \in A_i)Pr(q \in C_i))$$

$$+2(Pr(p \in B_i)Pr(q \in C_i)) \approx \frac{3}{8}$$

Remarque : dans le cas $i = 4$, $Pr(n \in ODD' \cap PPP^{(4)}) > \frac{3}{8}$

5.4.2 ... à celle de 2/3 des modules

Dans un premier temps, nous montrons la probabilité négligeable qu'un module appartienne à $\cap_{i=1}^{4} D_i$. L'ensemble D_i n'est pas couvert par le protocole généralisé : aucun des deux nombres premiers qui composent le module ne doit appartenir à D_i. Grâce à la combinaison des 4 protocoles, nous réduisons la proportion des modules dont l'un des facteurs appartient à l'intersection des ensembles D_i.

Dans un deuxième temps, on applique la probabilité précédente en prenant en compte cette nouvelle partie négligeable.

Afin de calculer la probabilité qu'un nombre premier p appartienne à l'intersection des ensembles D_i, il suffit de considérer le nombre de cas *interdits* modulo K_i donnés dans le Tableau 5.1 (noté L_i) sur le nombre de cas possibles modulo K_i (l'ensemble des nombres relativement premiers au ppcm des K_i) :

$$Pr(p \in \cap_{i \in I} D_i) = \frac{L_i}{\phi(ppcm(K_i, i \in I))}$$

I	$\{1\}$	$\{1,2\}$	$\{1,2,3\}$	$\{1,2,3,4\}$
$Pr(p \in \cap_{i \in I} D_i)$	1/4	1/8	$\frac{2}{32} = \frac{1}{16}$	$\frac{6}{192} = \frac{1}{32}$

TAB. 5.2 – Probabilité des cas non couverts

De la même façon que dans la partie précédente, on fait l'approximation suivante, en négligeant l'ensemble intersection $\cap_{i \in I} D_i$:

$$Pr(p \in \cup_{i \in I} A_i) \approx Pr(p \in \cup_{i \in I} B_i) \approx Pr(p \in \cup_{i \in I} C_i) \approx \frac{1}{3}$$

Ainsi, on obtient une approximation de la nouvelle couverture de modules considérés par la combinaison des 4 protocoles :

$$\begin{aligned} Pr(n \in ODD' \cap (\cup_{i \in I} PPP^{(i)})) \approx 2(Pr(p \in \cup_{i \in I} A_i) Pr(q \in \cup_{i \in I} B_i)) \\ +2(Pr(p \in \cup_{i \in I} A_i) Pr(q \in \cup_{i \in I} C_i)) \\ +2(Pr(p \in \cup_{i \in I} B_i) Pr(q \in \cup_{i \in I} C_i)) \approx \frac{2}{3} \end{aligned}$$

5.4.3 Apport d'information limité

Le résultat obtenu à travers la combinaison des protocoles, correspond à une amélioration de plus d'un quart de l'ensemble des modules composés de deux facteurs.

Néanmoins, le protocole n'est pas zero-knowledge. L'apport d'information provient de la donnée du nombre g_i. En réitérant plusieurs fois le protocole, le vérifieur détient de l'information sur le module, selon les ensembles liés aux nombres de base utilisés : celle-ci reste cependant limitée. En effet, un vérifieur malhonnête peut itérer plusieurs fois sa demande de validation du module d'un prouveur afin d'obtenir un ensemble de nombres de base qui conviennent (ceux-ci étant choisis au hasard à chaque preuve). L'information sur les facteurs qui composent le module est alors l'ensemble auquel appartient ce module : $\cap_{i \in I} PPP^{(i)}$. Il n'est pas évident cependant que la recherche

des nombres premiers en soit facilitée.

D'autre part, la prise en compte de l'ensemble des petits nombres premiers (codés par exemple sur un octet) comme nombre g_i engendrerait une couverture des modules composés de 2 nombres premiers qui se rapprocherait de la borne maximale de 2/3. Cependant, l'apport d'information serait plus important et ne justifierait peut-être pas le supplément minime de la couverture.

Conclusion

Cette thèse est destinée à mettre en lumière le principe de l'intégration du problème mathématique difficile de la factorisation des grands nombres, dans les protocoles de sécurité tels que GQ2 et Rabin-Williams.

Le premier chapitre rappelle les définitions utilisées en théorie des nombres, en particulier sur la factorisation des grands nombres. Les notions et le formalisme liés à la sécurité des protocoles d'authentification dynamique à 3 passes sont présentés. Un aperçu des protocoles d'authentification dont la sécurité se base sur le calcul difficile des racines successives, a mis en avant des techniques de construction, de complexité et de sécurité, identiques.

Le protocole GQ2 est né d'un besoin d'augmenter la sécurité des cartes bancaires. Dans les contextes d'intégration à fortes contraintes, ses performances permettent une authentification dynamique de l'ordre d'une seconde avec une taille de clé de 1024 bits. Une première analyse de sécurité conclut à des conditions de sécurité équivalentes à celles du problème difficile de la factorisation pour le quart des modules composés de 2 facteurs lorsque l'on utilise un seul nombre de base.

La généralisation du protocole GQ2 a permis de mettre en évidence la compatibilité du protocole avec une intégration RSA : la recherche des nombres de base affiche une probabilité écrasante de réussite pour un module composé de 2 facteurs quelconques. Le paramètre d'adaptation introduit est également utilisé pour valider, à partir d'une conjecture dans le cas général, les conditions de sécurité où l'on atteint un seuil de sécurité de la preuve de connaissance de la factorisation optimal. En particulier, dans la moitié des cas de modules, il suffit que le paramètre de sécurité soit égal au paramètre d'adaptation, lorsque l'on considère un seul nombre de base.

La généralisation à un module composé de 3, 4 ou 5 facteurs conserve un fort succès de réussite de cette recherche : plus de 9 chances sur 10 dans la majorité des cas. D'autre part, l'étude des performances liées à de plus larges exposants publics, a fait apparaître les limites de ces cas face à celles du protocole de Fiat-Shamir.

Les outils de représentation de corps utilisés pour la mise en place de cette version généralisée du protocole GQ2 à partir de la notion de niveau, ont été appliqués dans la nouvelle approche faite sur le schéma de signature de Rabin-Williams.

Les signatures électroniques de type Rabin doivent se protéger, de par leur équivalence avec la factorisation, contre certaines attaques, à l'aide de mécanismes de format. Nous avons extrait la structure des schémas de ce type pour justifier leur

construction qui répond à des exigences semblables. Ainsi, le nouveau schéma présenté regroupe et généralise les avantages de ces schémas, en permettant le maintien de l'utilisation d'un exposant de signature quel que soit le module considéré, avec la possibilité d'utiliser un exposant plus large.

Les entiers composés de grands facteurs premiers sont utilisés tout au long de ce document. La dernière partie est consacrée à l'étude des preuves non-interactives à divulgation nulle de connaissance qui attestent de la construction de ces modules. Une réunion de preuves, recouvrant chacune indépendamment 3/8 modules composés de 2 facteurs, permet une unique nouvelle preuve qui recouvre 2/3 des modules composés de 2 facteurs, avec un apport limité de connaissance.

Dans cette thèse, une nouvelle approche de la factorisation des grands nombres a été présentée à travers un théorème général de calcul de racines carrées d'ordre impair, quels que soient les facteurs du module considéré. Ces techniques ont été appliquées au protocole d'authentification GQ2 ainsi qu'au protocole de signature de Rabin-Williams, qui possèdent de bonnes performances et permettent le maintien du lien avec un problème mathématique toujours d'actualité après 25 ans de recherche intensive.

Annexe A

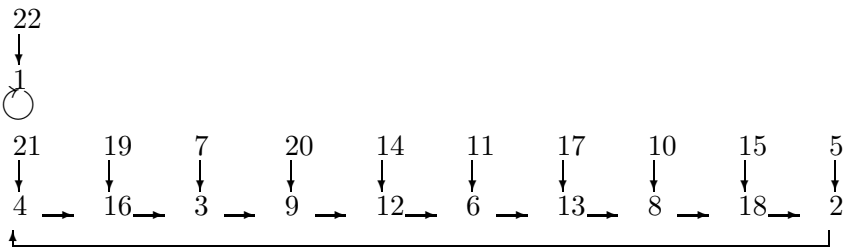
Annexes

A.1 Représentation graphique

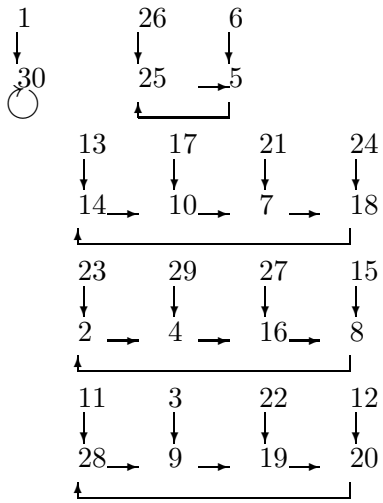
Cette première partie de l'annexe donne des exemples de représentation graphique du groupe multiplicatif $\mathbb{Z}_p^*$ où p est un nombre premier de première espèce pour la puissance e. Les cas $e = 2, 3, 4$ sont exposés.

Pour les résidus quadratiques $e = 2$

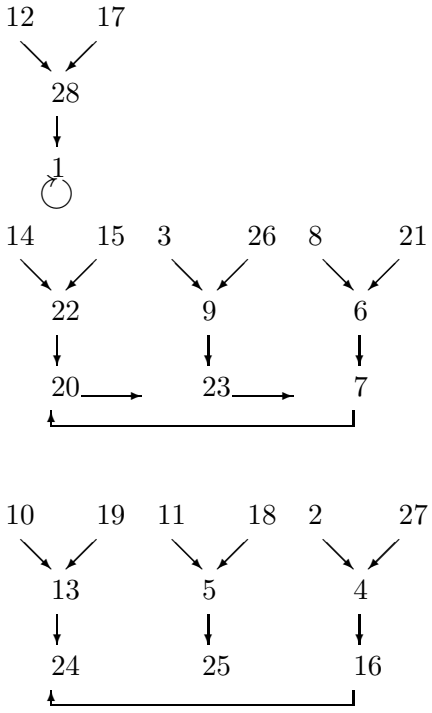
$p = 23 = 2.11 + 1$



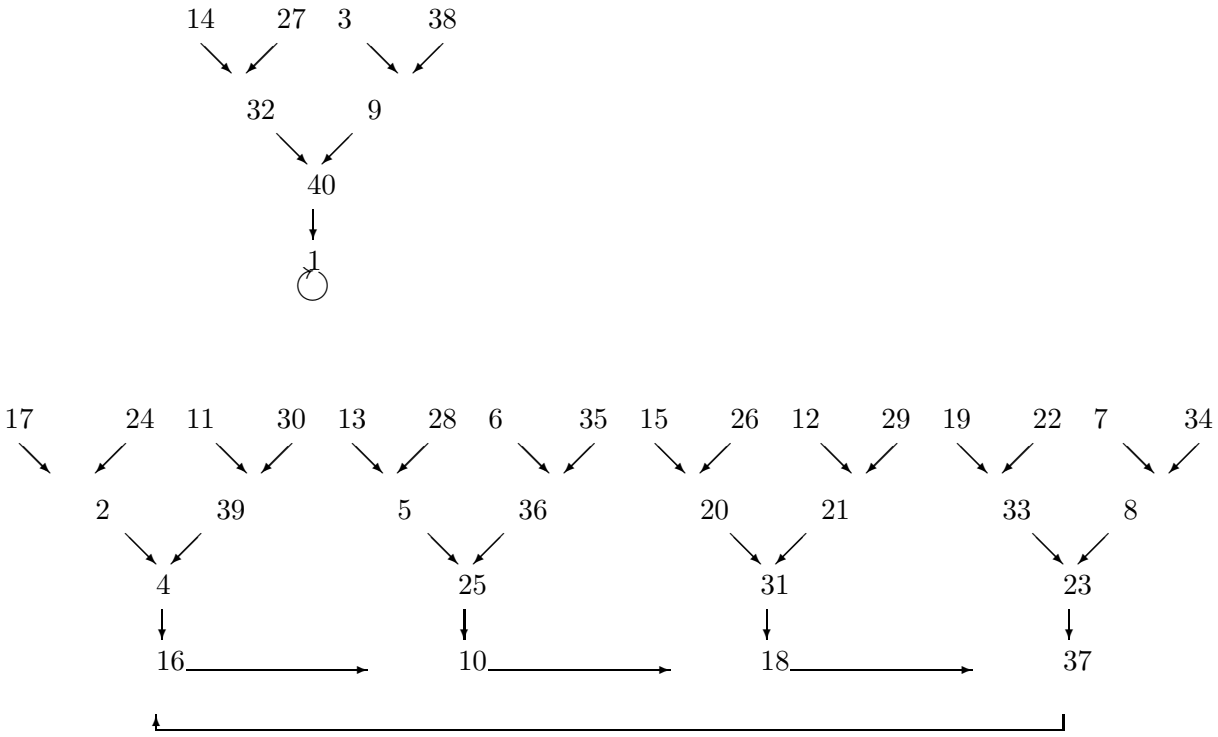
$p = 31 = 2.15 + 1$



$p = 29 = 2^2.7 + 1$

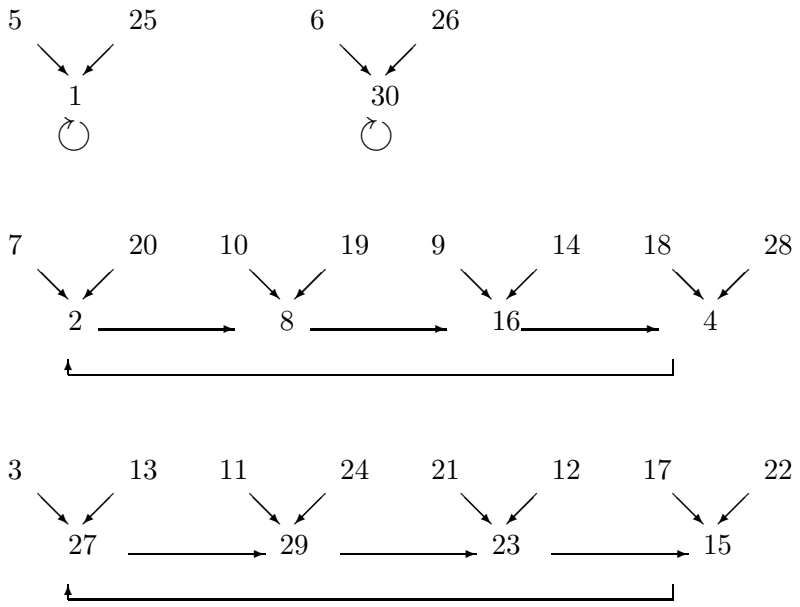


$p = 41 = 2^3.5 + 1$

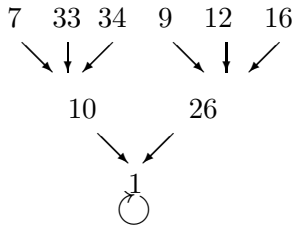


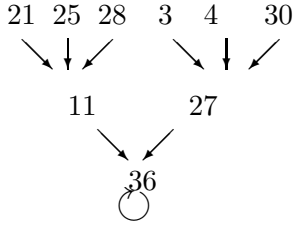
Pour les résidus cubiques $e = 3$

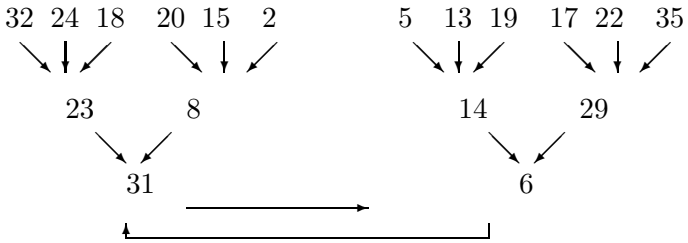
$p = 31 = 3^1 10 + 1$



$p = 37 = 3^2 4 + 1$

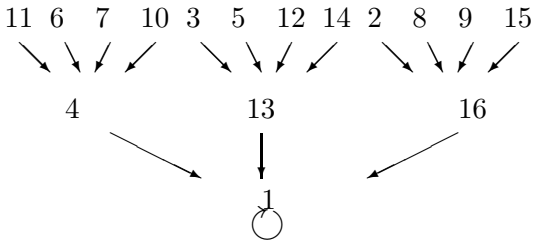




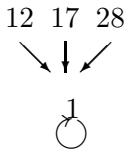


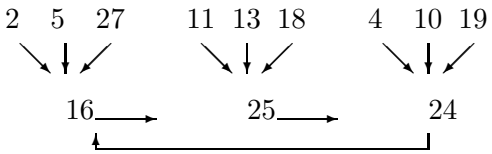
Pour les résidus biquadratiques $e = 4$

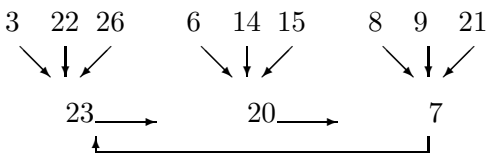
$p = 17 = 4^2 + 1$



$p = 29 = 4^1 7 + 1$







A.2 Complexités générales

Dans cette deuxième partie de l'annexe, on présente les diagrammes qui permettent de visualiser l'évolution de la complexité du protocole généralisé GQ2 en fonction des 3 paramètres suivants : le nombre de facteurs qui composent le module, la valeur du paramètre d'adaptation et la nature de l'exposant public.

Dans le cas de l'authentification faible ou forte, les résultats de complexité CPC (complexité du côté prouveur) et CPV (complexité du coté vérifieur) sont comparés à ceux du protocole de Fiat-Shamir à sécurité égale. Les colonnes pleines illustrent les avantages des complexités de GQ2 sur ce dernier.

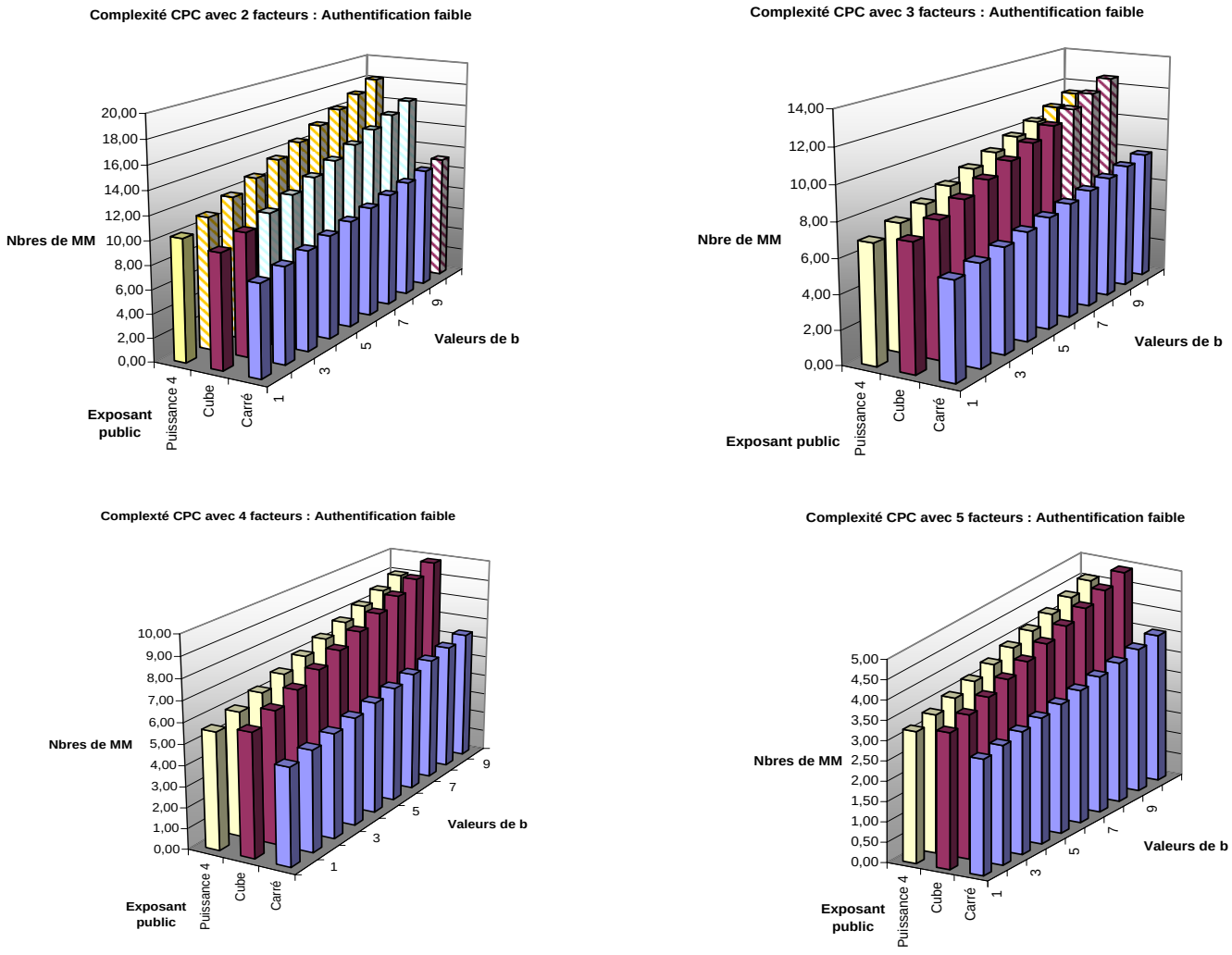


FIG. A.1 – CPC pour 2, 3, 4 et 5 facteurs : Authentification faible

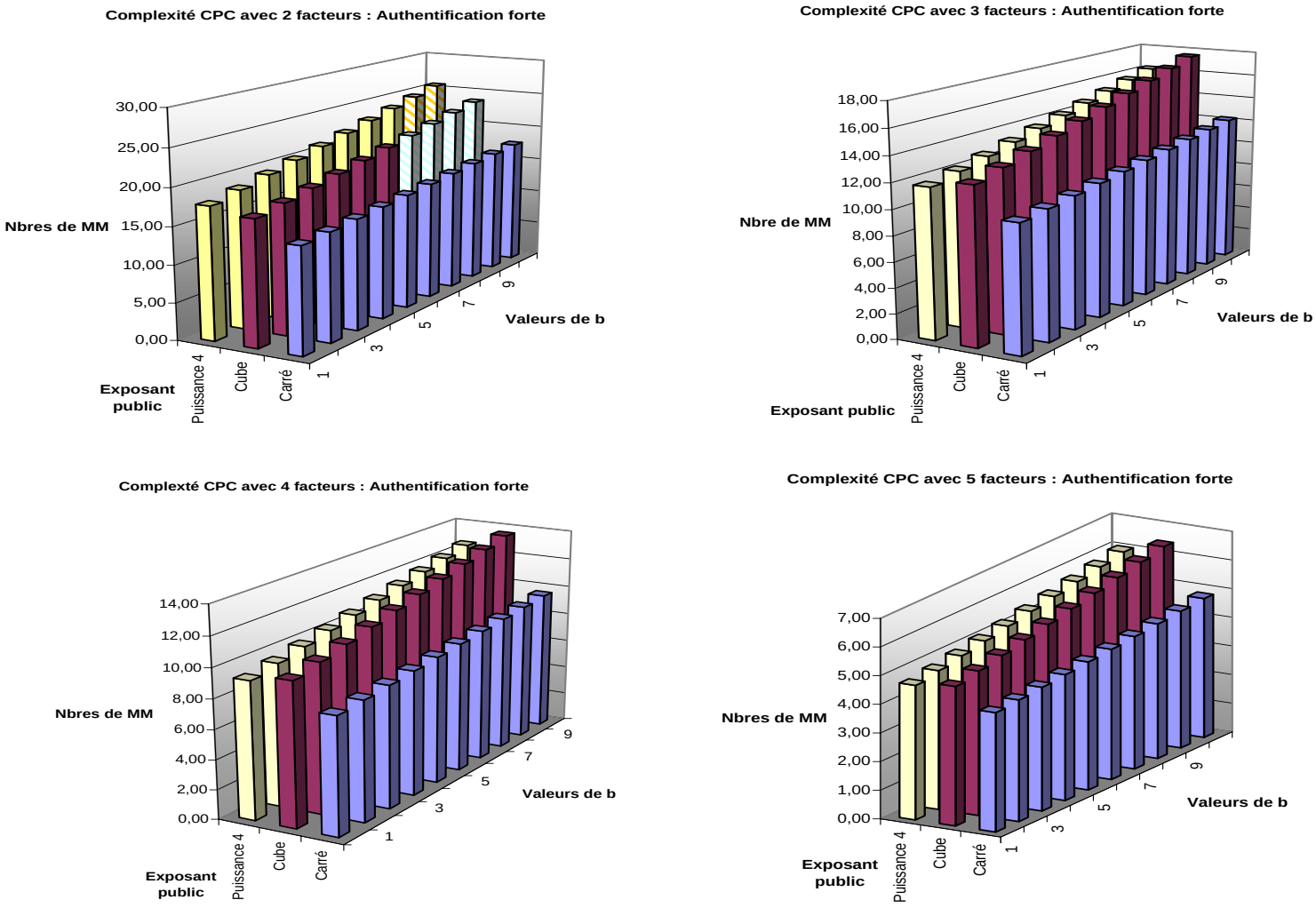


FIG. A.2 – CPC pour 2, 3, 4 et 5 facteurs : Authentification forte

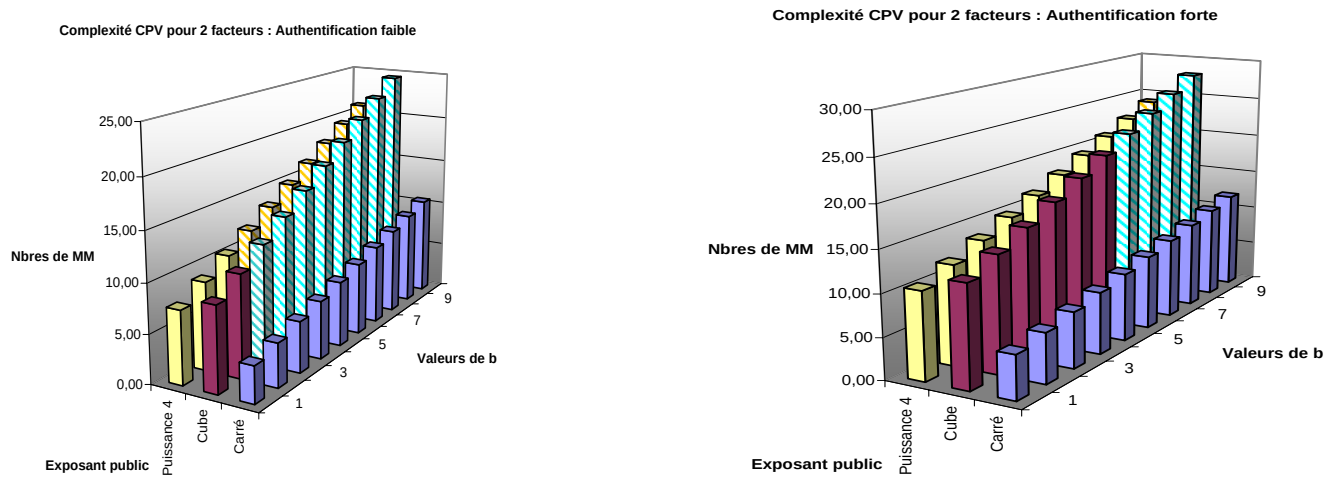


FIG. A.3 – CPV pour 2, 3, 4 et 5 facteurs : Authentification faible et forte

Bibliographie

[AKS02] Manindra Agrawal, Neeraj Kayal, and Nitin Saxena. PRIMES is in P, 2002.

[Arn02] François Arnault. Cours de cryptographie, 2002.

[BCJ+05] Eli Biham, Rafi Chen, Antoine Joux, Patrick Carribault, William Jalby, and Christophe Lemuet. Collisions of SHA-0 and reduced SHA-1. In *Proc. EUROCRYPT '05*, volume 3494 of *LNCS*, pages 36–57. Springer-Verlag, 2005.

[BDF98] Dan Boneh, Glenn Durfee, and Yair Frankel. An attack on RSA given a small fraction of the private key bits. In *Proc. ASIACRYPT '98*, volume 1514 of *LNCS*, pages 25–34. Springer-Verlag, 1998.

[BDG04a] Sophie Boutiton, François Daudé, and Louis Guillou. GQ2 un protocole zero-knowledge, complément essentiel à RSA. In *SAR'04*, pages 3–11, 2004.

[BDG04b] Sophie Boutiton, François Daudé, and Louis Guillou. GQ2 une preuve zero-knowledge de connaissance de la factorisation. In *SSTIC'04*, pages 116–128, 2004.

[BDL97] Dan Boneh, Richard A. Demillo, and Richard J. Lipton. On the importance of checking cryptographic protocols for faults. In *Proc. EUROCRYPT '97*, volume 1233 of *LNCS*, pages 37–51. Springer-Verlag, 1997.

[Ber01] Daniel J. Bernstein. Circuits for integer factorization : A proposal, November 10 2001.

[Berre] Daniel Bernstein. A secure public-key signature system with extremely fast verication. *Journal of Cryptology*, à paraître.

[BFL94] Carsten Boyar, Joan Friedl, and Katalin Lund. Practical zero-knowledge proofs : giving hints and using deficiencies. Technical report, University of Chicago, 1994.

[Blu81] Manuel Blum. Coin flipping by telephone : A protocol for solving impossible problems. In *Proc. CRYPTO '81*, pages 11–15. Department of Electrical and Computer Engineering, U. C. Santa Barbara, 1981.

[BM03] Johannes Blömer and Alexander May. New partial key exposure attacks on RSA. In *Proc. CRYPTO '03*, volume 2729 of *LNCS*, pages 27–43. Springer-Verlag, 2003.

[BNN04] Mihir Bellare, Chanathip Namprempre, and Gregory Neven. Security proofs for identity-based identification and signature schemes. In *Proc. EUROCRYPT '04*, volume 3027, pages 268–286. Springer-Verlag, 2004.

[BP02] Mihir Bellare and Adriana Palacio. GQ and Schnorr identification schemes : Proofs of security against impersonation under active and concurrent attacks. In *Proc. CRYPTO '02*, volume 2442 of *LNCS*, pages 162–178. Springer-Verlag, 2002.

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical : A paradigm for designing efficient protocols. *ACM Conference on Computer and Communications Security*, pages 62–73, 1993.

[BR96] Mihir Bellare and Phillip Rogaway. The exact security of digital signatures — how to sign with RSA and Rabin. In *Proc. EUROCRYPT '96*, volume 1070 of *LNCS*, pages 399–416. Springer-Verlag, 1996.

[BSMP91] Manuel Blum, Alfredo De Santis, Silvio Micali, and Giuseppe Persiano. Non-interactive zero-knowledge. *SIAM Journal on Computing*, 20(6) :1084–1118, 1991.

[BV98] Dan Boneh and Ramarathnam Venkatesan. Breaking RSA may not be equivalent to factoring. In *Proc. EUROCRYPT '98*, volume 1233 of *LNCS*, pages 59–71. Springer-Verlag, 1998.

[CEvdG87] David Chaum, Jan-Hendrik Evertse, and Jeroen van de Graaf. An improved protocol for demonstrating possession of discrete logarithms and some generalizations. In *Proc. EUROCRYPT 87*, volume 304 of *LNCS*, pages 127–141. Springer-Verlag, 1987.

[CGH98] Ran Canetti, Oded Goldreich, and Shai Halevi. The random oracle methodology, revisited (preliminary version). In *Proc. ACM Symposium on Theory of Computing*, pages 209–218. ACM Press, 1998.

[CGP01] Nicolas Courtois, Louis Goubin, and Jacques Patarin. Flash, a fast multivariate signature algorithm. In *Progress in Cryptology - CT-RSA 2001*, volume 2020 of *LNCS*, pages 298–307. Springer-Verlag, 2001.

[CM99] Jan Camenisch and Markus Michels. Proving in zero-knowledge that a number is the product of two safe primes. In *Proc. EUROCRYPT ' 99*, volume 1599 of *LNCS*, pages 107–122. Springer-Verlag, 1999.

[CM04] Jean-Sebastien Coron and Alexander May. Deterministic polynomial time equivalence of computing the RSA secret key and factoring, 2004.

[CNS99] Jean-Sébastien Coron, David Naccache, and Julien P. Stern. On the security of RSA padding. In *Proc. CRYPTO '99*, volume 1666 of *LNCS*, pages 1–18, 1999.

[Coh93] Henri Cohen. *A course in computational algebraic number theory*, volume 138 of *Graduate Texts in Mathematics*. Springer-Verlag, 1993.

[Cor00] Jean-Sébastien Coron. On the exact security of full domain hash. In *Proc. CRYPTO '00*, volume 1880 of *LNCS*, pages 229–235. Springer-Verlag, 2000.

[Cor01] Jean-Sébastien Coron. *Cryptanalyses et preuves de sécurité de schémas à clé publique*. PhD thesis, Ecole Nationale supérieure, 2001.

[DBP96] Hans Dobbertin, Antoon Bosselaers, and Bart Preneel. RIPEMD-160 : A strengthened version of RIPEMD. In *FSE'96*, volume 1039 of *LNCS*, pages 71–82. Springer-Verlag, 1996.

[Déf80] Marc Défourneaux. *Do you speak science ? Comment s'exprimer en anglais scientifique*. Gauthier-Villars, 1980.

[DH76] Whitfield Diffie and Martin Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6) :644–654, 1976.

[dJC85] Wiebren de Jonge and David Chaum. Attacks on some RSA signatures. In *Proc. CRYPTO '85*, volume 218 of *LNCS*, pages 18–27. Springer-Verlag, 1985.

[DO86] Y. Desmedt and A. M. Odlyzko. A chosen text attack on the RSA cryptosystem and some discrete logarithm schemes. In *Proc. CRYPTO '85*, volume 218 of *LNCS*, pages 516–522. Springer-Verlag, 1986.

[dVC01] Françoise Lévy dit Véhel and Anne Canteaut. La cryptographie moderne. *L'Armement*, 2001.

[EJMdW05] Matthias Ernst, Ellen Jochemsz, Alexander May, and Benne de Weger. Partial key exposure attacks on RSA up to full size exponents. In *Proc. EUROCRYPT '05*, volume 3494 of *LNCS*, pages 371–386. Springer-Verlag, 2005.

[ElG85] Taher ElGamal. A public key and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4) :469–472, 1985.

[FF02] Marc Fischlin and Roger Fischlin. The representation problem based on factoring. In *CT-RSA*, LNCS, 2002.

[FFS89] Uriel Feige, Amos Fiat, and Adi Shamir. Zero knowledge proofs of identity. In *Proc. CRYPTO '89*, volume 435 of *LNCS*, pages 526–544, 1989.

[FS87] Amos Fiat and Adi Shamir. How to prove Yourself : Practical solutions of identification and signature problems. In *Proc. CRYPTO '86*, volume 263 of *LNCS*, pages 186–194, 1987.

[Gir01] Marc Girault. Self-certified public keys. In *Proc. EUROCRYPT '91*, volume 36 of *LNCS*, pages 437–451, 2001.

[GK03] Shafi Goldwasser and Yael Kalai. On the (In)security of the Fiat-Shamir paradigm. In *Proc. IEEE Symposium on Foundations of Computer Science*, pages 102–113, 2003.

[GM97] Marc Girault and Jean François Misarsky. Selective forgery of RSA signatures using redundancy. In *Proc. EUROCRYPT '97*, volume 1233 of *LNCS*, pages 495–507, 1997.

[GMR88] Shafi Goldwasser, Silvio Micali, and Ron L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2) :281–308, 1988.

[GMR89] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1) :186–208, February 1989.

[GMR98] Rosario Gennaro, Daniele Micciancio, and Tal Rabin. A efficient non-interactive statistical zero-knowledge proof system for quasi-safe prime products. *ACM Conference on Computer and Communications Security*, pages 67–72, 1998.

[Gol01] Oded Goldreich. *Foundations of Cryptography*, volume I. Cambridge University Press, 2001.

[Gol02] Oded Goldreich. Zero-knowledge twenty years after its invention. Technical report, U.S.C. Computer Science Department, 2002.

[GP88] Jeroen Van De Graaf and René Peralta. A simple and secure way to show the validity of your public key. In *Proc. CRYPTO'87*, volume 293 of *LNCS*, pages 128–134, 1988.

[GQ88a] Louis C. Guillou and J.-J. Quisquater. A "paradoxical" identity-based signature scheme resulting from zero-knowledge. In *Proc. CRYPTO '88*, volume 403 of *LNCS*, pages 216–231. Springer-Verlag, 1988.

[GQ88b] Louis C. Guillou and Jean-Jacques Quisquater. A practical zero-knowledge protocol fitted to security microprocessor minimizing both transmission and memory. In *Proc. EUROCRYPT'88*, volume 330 of *LNCS*, pages 123–128, 1988.

[GQ90] Louis C. Guillou and Jean-Jacques Quisquater. How to explain zero-knowledge protocols to Yours children. In *Proc. CRYPTO'89*, volume 435 of *LNCS*, pages 628–631, 1990.

[GQ01] Louis C. Guillou and Jean-Jacques Quisquater. Technical report on dynamic authentication comparaison of RSA, GQ1 and GQ2. Technical report, France Telecom R&D/ University of Louvain, 2001.

[GQU01] Louis C. Guillou, Jean-Jacques Quisquater, and Michel Ugon. Cryptographic authentication protocols for smarts cards. *Computer Network Magazine*, 36 :437–451, 2001.

[Gui04] Louis C. Guillou. Histoire de la carte à puce du point de vue d'un cryptographe. In *Histoire de l'Informatique et des Télécommunications*. ESAT, 2004.

[HPS98] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU : A ring-based public key cryptosystem. In *Algorithmic Number Theory*, volume 1423 of *LNCS*, pages 267–288, 1998.

[IR82] Kenneth Ireland and Michael Rosen. *A Classical Introduction to Modern Number Theory*, volume 84 of *Graduate Texts in Mathematics*. Springer-Verlag, 1982.

[ISO04] ISO/IEC-9798-5. Information technology - security techniques . entity authentication . part 5 : Mechanisms using zero-knowledge techniques, 2004.

[ISOre] ISO/IEC-14888-2. Information technology - security techniques . digital signature appendix . part 2 : Integer factorization based mechanisms, A paraître.

[JJK99] Benjamin Jun, Joshua Jaffe, and Paul Kocher. Differential power analysis. In *Proc. CRYPTO '99*, volume 1666 of *LNCS*, pages 388–397. Springer-Verlag, 1999.

[Joy97] Marc Joye. *Security analysis of RSA-type cryptosystems*. PhD thesis, Université Catholique de Louvain, 1997.

[KIT88] Kaoru Kurosawa, T. Ito, and M. Takeuchi. Public key cryptosystem using a reciprocal number with the same intractability as factoring a large number. *Cryptologia*, XII(4) :225–233, 1988.

[KO99] Kaoru Kurosawa and Wakaha Ogata. Efficient Rabin-type digital signature scheme. In *DCC*, volume 16, pages 53–64, 1999.

[Kob87] Neal Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48(177) :203–209, January 1987.

[Koc96] Paul C. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In *Proc. CRYPTO '96*, volume 1109 of *LNCS*, pages 104–113. Springer-Verlag, 1996.

[KOMM01] Kaoru Kurosawa, W. Ogata, Toshihiko Matsuo, and Shuichi Makishima. IND-CCA public key schemes equivalent to factoring n=pq. In *Proc. PKC '01*, LNCS, page 36, 2001.

[Len80] H. W. Lenstra. *Euclidean number fields*. Number 2 in The Mathematical Intelligencer. Springer-Verlag, 1980.

[LKBS92] J. H. Loxton, David S. P. Khoo, Gregory J. Bird, and Jennifer Seberry. A cubic RSA code equivalent to factorization. *Journal of Cryptology*, 5(2) :139–150, 1992.

[LLL82] A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261 :513–534, 1982.

[LS98] Moses Liskov and Robert D. Silverman. A statistical limited-knowledge proof for secure RSA keys. *IEEE P1363 Research Contributions*, 1998.

[LSTT02] Arjen K. Lenstra, Adi Shamir, Jim Tomlinson, and Eran Tromer. Analysis of Bernstein 's factorization circuit. In *Proc. ASIACRYPT'02*, volume 2501 of *LNCS*, pages 1–26. Springer-Verlag, 2002.

[May04] Alexander May. Computing the RSA secret key is deterministic polynomial time equivalent to factoring. In *Proc. CRYPTO '04*, volume 3152 of *LNCS*, pages 213–219. Springer-Verlag, 2004.

[Mic94] Silvio Micali. A secure and efficient digital signature algorithm. Technical Memo MIT/LCS/TM-501b, Messachusetts Institute of Technology, 1994.

[Mil86] Victor Miller. Use of elliptic curves in cryptography. In *Proc. CRYPTO '85*, volume 218 of *LNCS*, pages 417–426, Berlin, 1986. Springer-Verlag.

[Mis98] Jean-François Misarsky. How (not) to design RSA signature schemes. In *Proc. PKC '98*, volume 1431 of *LNCS*, pages 14–28. Springer-Verlag, 1998.

[MOV97] Alfred Menezes, Paul C.van Oorschot, and Scott Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1997.

[MS88] S. Micali and A. Shamir. An improvement of the Fiat-Shamir identification and signature scheme. In *Proc. CRYPTO '88*, volume 403 of *LNCS*. Springer-Verlag, 1988.

[Oka92] Tatsuaki Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. In *Proc. CRYPTO '92*, volume 740 of *LNCS*, pages 31–53. Springer-Verlag, 1992.

[OO88] Kazuo Ohta and Tatsuaki Okamoto. A modification of the Fiat-Shamir scheme. In *Proc. CRYPTO '88*, volume 403 of *LNCS*, pages 232–243. Springer-Verlag, 1988.

[OS91] H. Ong and Claus-Peter Schnorr. Fast signature generation with a Fiat-Shamir-like scheme. In *Proc. EUROCRYPT'90*, volume 473 of *LNCS*, pages 432–440, 1991.

[Pat96] Jacques Patarin. Hidden fields equations (HFE) and Isomorphisms of Polynomials (IP) : two new families of asymmetric algorithms. In *Proc. EUROCRYPT '96*, volume 1070, pages 33–48. Springer-Verlag, 1996.

[Pat02] Jacques Patarin. L'art du secret. *Dossier pour la science*, 36 :66, juillet/octobre 2002.

[Poi96] David Pointcheval. *Les preuves de connaissance et les preuves de sécurité.* PhD thesis, Ecole Nationale Supérieure, 1996.

[Pou00] Guillaume Poupard. *Authentification d'entités, de messages et de clés cryptographiques : théorie et pratique.* PhD thesis, Ecole Nationale Supérieure, 2000.

[PS96] David Pointcheval and Jacques Stern. Security proofs for signature schemes. In *Proc. EUROCRYPT'96*, volume 1070, pages 387–398, 1996.

[PS98] Guillaume Poupard and Jacques Stern. Security analysis of a practical «on the fly »authentication and signature generation. In *Proc. EUROCRYPT '98*, volume 1403 of *LNCS*, pages 422–436, 1998.

[PS00] Guillaume Poupard and Jacques Stern. Short proofs of knowledge for factoring. In *Proc. PKC'00*, volume 1751 of *LNCS*, pages 147–166. Springer-Verlag, 2000.

[QJ01] Jean-Jacques Quisquater and Marc Joye. On Rabin-type signatures. In *WCC*, volume 2260 of *LNCS*, pages 99–113. Springer-Verlag, 2001.

[Rab79] Mickaël O. Rabin. Digitalized signatures and public-key functions as intractable as factorization. Technical report, Massachusetts Institute of Technology, 1979.

[Riv91] Ronald L. Rivest. The MD4 message digest algorithm. In *Proc. CRYPTO '90*, volume 537, pages 303–311. Springer-Verlag, 1991.

[Riv92] Ronald L. Rivest. The MD5 message-digest algorithm, 1992.

[Rol99] Christian Rolland. *Latex par la pratique.* O'Reilly, 1999.

[RSA78] Ron L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public key cryptosytems. *ACM Conference on Computer and Communications Security*, 21(2) :120–126, 1978.

[Sch89] C. P. Schnorr. Efficient identification and signatures for smart cards. In *Proc. EUROCRYPT '89*, volume 434 of *LNCS*, pages 688–689. Springer-Verlag, 10–13 April 1989.

[Sch94] Bruce Schneier. *Applied Cryptography.* Wiley, 1994.

[Sch97] C. P. Schnorr. Security of 2^t-root identification and signatures. In *Proc. CRYPTO'97*, volume 1294 of *LNCS*, page 540, 1997.

[Sch98] Renate Scheidler. A public-key cryptosystem using purely cubic fields. *Journal of Cryptology*, 11(2) :109–124, 1998.

[Sha84] Adi Shamir. Identity-based cryptosystems and signature schemes. In *Proc. CRYPTO 84*, volume 196 of *LNCS*, pages 47–53. Springer-Verlag, 1984.

[Sho96] Victor Shoup. On the security of a practical identification scheme. In *Proc. EUROCRYPT'96*, volume 1070 of *LNCS*, pages 340–353, 1996.

[Sho04] Victor Shoup. A computational introduction to number theory and algebra, 2004.

[Sti95] Douglas Stinson. *Cryptographie : Théorie et pratique*. CRC Press, 1995.

[SW95] Renate Scheidler and Hugh C. Williams. A public-key cryptosystem utilizing cyclotomic fields. *DCC*, 6(2) :117–131, 1995.

[Wil80] H. C. Williams. A modification of the RSA public-key encryption procedure. *IEEE Transactions on Information Theory*, 26(6) :726–729, 1980.

[Wil84] H. C. Williams. Some public key crypto-functions as intractable as factorization. In *Proc. CRYPTO 84*, volume 196 of *LNCS*, pages 66–70. Springer-Verlag, 1984.

[Wil85] H. C. Williams. An M^3 public-key encryption scheme. In *Proc. CRYPTO 85*, volume 218 of *LNCS*, pages 358–368. Springer-Verlag, 1985.

[WLF+05] Xiaoyun Wang, Xuejia Lai, Dengguo Feng, Hui Chen, and Xiuyuan Yu. Cryptanalysis of the hash functions MD4 and RIPEMD. In *Proc. EUROCRYPT '05*, volume 3494 of *LNCS*, pages 1–18. Springer-Verlag, 2005.

[WY05] Xiaoyun Wang and Hongho Yu. How to break MD5 and other hash functions. In *Proc. EUROCRYPT '05*, LNCS, pages 19–35. Springer-Verlag, 2005.

Résumé

L'objectif de cette thèse est de généraliser les protocoles de sécurité GQ2 et Rabin-Williams, tous deux reliés au problème difficile de la factorisation des grands nombres.

Nous évaluons la sécurité du schéma GQ2, en particulier à travers l'estimation de la capacité d'usurper une identité relativement à la capacité de factoriser la clé publique. Puis, nous montrons la forte probabilité de générer des clés GQ2 compatibles avec l'utilisation de modules RSA généraux, de modules multi-facteurs, ou de plus grands exposants publics.

Dans le domaine de la signature, aucun schéma de type Rabin-Williams n'a jamais réussi à concilier l'utilisation d'un exposant unique de signature quel que soit le module considéré. Nous proposons ici une solution qui généralise naturellement les précédents schémas.

Le dernier chapitre reprend une technique de preuve de validation du module. Sans améliorer les résultats antérieurs, un protocole couvrant une classe de modules de forme particulière est présenté.

Summary

This thesis is about the generalization of the GQ2 and Rabin-Williams schemes. Their security relies on the problem of factorization of large numbers.

First we have a look at the security of the GQ2 protocol by giving an estimation of the power of cheating related to the power of factorizing the public key number. Then we prove the overwhelming probability to generate GQ2 keys from general RSA moduli, from moduli composed by more than two factors, or from larger public exponents.

Up until now, none of Rabin-Williams type digital signatures have a unique signature exponent when considering any two-factor modulus. We present here a solution that also generalizes the previous schemes.

The last chapter is based on an article by Gennaro, Micciancio and Rabin (CCS99). Their results are not improved here but a new proof is given, using technical tools, to prove the validity of some two-factor moduli.