Code-based Identification and Signature Schemes

Vom Fachbereich Informatik der
Technischen Universität Darmstadt genehmigte

Dissertation

zur Erlangung des Grades
Doktor rerum naturalium (Dr. rer. nat.)

von

Dipl.-Math. Sidi Mohamed El Yousfi Alaoui

geboren in Ksar Jramna, Marokko.

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Referenten:	Prof. Dr. Johannes Buchmann Assoc. Prof. Dr. Pierre-Louis Cayrel Prof. Dr. Ayoub Otmani
Tag der Einreichung:	01. Mai 2012
Tag der mündlichen Prüfung:	03. June 2013
Hochschulkennziffer:	D 17

Darmstadt 2013

Wissenschaftlicher Werdegang

März 2009 - heute

Promotionsstudent am Lehrstuhl von Prof. Johannes Buchmann, Fachbereich Informatik, — Kryptographie und Computeralgebra, Technische Universität Darmstadt.

Oktober 2002 - September 2007

Studium der Mathematik mit Schwerpunkt Informatik an der Technischen Universität Darmstadt.

List of Publications

[1] El Yousfi Alaoui, S. M., Özgür, D., Véron, P., Galindo, D., Cayrel, P.-L.: Extended Security Arguments for Signature Schemes. The 5th African International Conference on Cryptology, AfricaCrypt 2012, LNCS 7374, pages 19-34, Springer, 2012.

[2] Cayrel, P.-L., El Yousfi Alaoui, S. M., Véron, P., Hoffmann, G.: An Improved Threshold Ring Signature Scheme based on Error-Correcting Codes. International Workshop on the Arithmetic of Finite Fields, WAIFI 2012, LNCS 7374, pages 54-63, Springer, 2012.

[3] Hülsing, A., Petzoldt, A., Schneider, M., El Yousfi Alaoui, S. M.: Postquantum Signaturverfahren Heute. In: Waldmann, Ulrich (Editor): 22. SIT-Smartcard Workshop 2012, Fraunhofer Verlag Stuttgart, Februar 2012. ISBN 978-3-8396-0347-5.

[4] Meziani, M., El Yousfi Alaoui, S. M., Cayrel, P.-L.: Hash Functions based on Coding Theory. Proceedings of the 2nd Workshop on Codes, Cryptography and Communication Systems WCCCS 2011, pages 32-37, June 2011, Rabat, Morocco.

[5] Meziani, M., Cayrel, P.-L., El Yousfi Alaoui, S. M.: 2SC: An Efficient Code-based Stream Cipher. The 5th International Conference on Information Security and Assurance, ISA 2011, Volume 200, pages 111-122, Springer, 2011.

[6] Meziani, M., Özgür, D., Cayrel, P.-L., El Yousfi Alaoui, S. M.: S-FSB: An improved Variant of the FSB Hash Family. The 5th International Conference on Information Security and Assurance, ISA 2011, Volume 200, pages 132-145, Springer, 2011.

[7] Cayrel, P.-L., El Yousfi Alaoui, S. M., Hoffmann, G., Meziani, M., Niebuhr, R.: Recent Progress in Code-based Cryptography. The 5th International Conference on Information Security and Assurance, ISA 2011, Volume 200, pages 21-32, Springer, 2011.

[8] El Yousfi Alaoui, S. M., Cayrel, P.-L., Meziani, M.: Improved Identity-based Identification and Signature Schemes using Quasi-Dyadic Goppa Codes, The 5th International Conference on Information Security and Assurance, ISA 2011, Volume 200, pages 146-155, Springer, 2011.

[9] Cayrel, P.-L., El Yousfi Alaoui, S. M., Günther, F., Hoffmann, G., Rothe, H.: Efficient Implementation of Code-based Identification and Signature Schemes, Western European Workshop on Research in Cryptology, WEWoRC 2011, July 2011, Weimar, Germany.

[10] Cayrel, P.-L., El Yousfi Alaoui, S. M.: Dual Construction of Stern-based Signature Schemes. Proceedings of the International Conference on Cryptography, Coding and Information Security, ICCCIS 2010, Volume 63, March 2010 ISSN 2070-3724, Rio de Janeiro, Brasil, 2010.

[11] Cayrel, P.-L., Véron, P., El Yousfi Alaoui, S. M.: A Zero-Knowledge Identification Scheme based on the q-ary Syndrome Decoding Problem, Selected Areas in Cryptography, SAC 2010, LNCS 6544, pages 171-186, Springer, 2010.

Acknowlegdement

At the end of my thesis I would like to thank all those people who have contributed in many ways for making this thesis possible.

First of all I would like to express my sincere gratitude to Johannes Buchmann for giving me the opportunity to write this thesis under his systematic guidance. His patience and encouragement helped me during all time of my research.

Furthermore I would like to extend my deepest gratitude to Pierre-Louis Cayrel, for his supervision, numerous contributions, his crucial advices and for his constant willingness to share his knowledge and savoir faire during my work on this thesis. I am especially grateful to Ayoub Otmani for agreeing to be my co-referee.

Very specials thanks to Özgür Dagdelen, David Galindo, Gerhard Hofmann, Andreas Hülsing, Meziani Mohammed, Albrecht Petzoldt, Michael Schneider, and Pascal Véron for the interesting and beneficial joint work.

I also owe very much gratitude to the Center for Advanced Security Research Darmstadt (CASED) for the financial support.

I warmly thank all CDC and CASED members for the motivating working atmosphere, especially to Sami Alsouri, Mohamed Saied Emam Mohamed, and Rachid El Bansarkhani. I would also like to thank my friends Mohammed Naoufal Adraoui, Omar Bakhcha, Younes Bennani, Aboulkacime Chebak, Youness Marhraoui, Khalid Mouaouine, Sheikh Amine Elhalouti, Mohsine Ghazali, Jalal hbirkou, Abdelali Ramh, and others. They were always supporting me and encouraging me with their best wishes.

Last but not least, I would like to thank my parents, sisters, and brothers for their support and encouragement from a long distance. To my wife for her affection, her great understanding and sacrifice during the last years. To my beloved daughter Oumaima and son Anass, I am sorry that I left them without my care even they are too young.

Darmstadt,
April 2013

Sidi Mohamed El Yousfi Alaoui

Abstract

In an age of explosive growth of digital communications and electronic data storage, cryptography plays an integral role in our society. Some examples of daily use of cryptography are software updates, e-banking, electronic commerce, ATM cards, etc. The security of most currently used cryptosystems relies on the hardness of the factorization and discrete logarithm problems. However, in 1994 Peter Shor discovered polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. Therefore, it is of extreme importance to develop cryptosystems that remain secure even when the adversary has access to a quantum computer; such systems are called *post-quantum cryptosystems*. One promising candidate is based on codes; in this thesis we focus more specifically on code-based identification and signature schemes.

Public key identification schemes are typically applied in cryptography to reach the goal of entity authentication. Their applications include authentication and access control services such as remote login, credit card purchases and many others. One of the most well-known systems of this kind is the zero-knowledge identification scheme introduced in Crypto 1993 by Stern. It is very fast compared to schemes based on number-theoretic problems since it involves only simple and efficiently executable operations. However, its main drawbacks are the high communication complexity and the large public key size, that makes it impractical for many applications.

Our first contribution addresses these drawbacks by taking a step towards reducing communication complexity and public key size simultaneously. To this end, we propose a novel zero-knowledge five-pass identification scheme which improves on Stern's scheme. It reduces the communication complexity by a factor of 25% compared to Stern's one. Moreover, we obtain a public key of size of 4 KB, whereas Stern's scheme requires 15 KB for the same level of security. To the best of our knowledge, there is no code-based identification scheme with better performance than our proposal using random codes. Our second contribution consists of extending one of the most important paradigms in cryptography, namely the one by Fiat and Shamir. In doing so, we enlarge the class of identification schemes to which the Fiat-Shamir transform can be applied. Additionally, we put forward a generic methodology for proving the security of signature schemes derived from this class of identification schemes. We exemplify our extended paradigm and derive a provably secure signature scheme based on our proposed five-pass identification scheme. In order to contribute to the development of post-quantum schemes with additional features, we present an improved code-based threshold ring signature scheme using our two previous results. Our proposal has a shorter signature length and a smaller public-key size compared to Aguilar et al.'s scheme, which is the reference in this area.

Zusammenfassung

Gegenwärtig spielt die Kryptographie eine fundamentale Rolle bei der Absicherung einer Vielzahl von täglichen Anwendungen und Prozessen. Dazu gehören beispielsweise Software-Updates, E-Commerce und E-Banking Anwendungen. Die Sicherheit der am häufigsten in der Praxis eingesetzten kryptographischen Verfahren beruht auf der Schwierigkeit, große Zahlen in ihre Primfaktoren zu zerlegen oder diskrete Logarithmen zu berechnen. Im Jahr 1994 präsentierte Peter Shor Algorithmen, mit denen das Problem der Faktorisierung und des diskreten Logarithmus in polynomieller Laufzeit mittels Quantencomputern gelöst werden können. Daher ist es von äußerster Bedeutung nach Alternativen zu suchen, die langfristig als Ersatz fungieren. Als mögliche Kandidaten kommen Code-, gitter-, multivariate-, und hash-basierte Kryptosysteme in Betracht, die in den letzten Jahren große Erfolge verzeichnen konnten. Die Untersuchung und das Design von Code-basierten Identifikation- und Signaturverfahren stellen den Kerninhalt dieser Arbeit dar.

Im Jahr 1993 wurde von Jacques Stern das erste effiziente Identifikationsverfahren veröffentlicht, welches auf Codierungstheorie basiert. Es hat allerdings zwei Nachteile, welche die Größe des öffentlichen Schlüssels und die hohen Kommunikationskosten betreffen. Zu diesem Zweck schlagen wir ein neues 5-Pass Zero-Knowledge-Identifikationsverfahren vor, das unseres Wissens nach alle Code-basierten Verfahren übertrifft. Mittels unserer Konstruktion werden die Kommunikationskosten um bis zu 25% und die Größe des öffentlichen Schlüssels von 15 KB auf 4 KB im Vergleich zum Verfahren von Stern reduziert. Als weiteres Ergebnis präsentieren wir eine Verallgemeinerung der Fiat-Shamir Heuristik, welches eines der wichtigsten Paradigmen in der Kryptographie darstellt. Diese wird dazu verwendet, um aus einem kanonischen (3-Pass) Identifikationsprotokoll ein Signaturverfahren zu konstruieren. Ebenfalls entwickeln wir einen Sicherheitsbeweis für diese Transformation für Protokolle, die nicht kanonischen sind. Mit dieser Verallgemeinerung kann man nun unsere Identifikationsverfahren als auch viele andere nicht kanonischen Identifikationsprotokolle zu sicheren Signaturverfahren transformieren. Im letzten Teil schlagen wir ein Threshold Ring Signaturverfahren vor, ein Signaturverfahren mit speziellen Eigenschaften, dem das vorgeschlagene Zero-Knowledge-Identifikationsverfahren zugrunde liegt. Es stellt sich heraus, dass unsere Konstruktion in Bezug auf die Signaturlänge, die Größe des öffentlichen Schlüssels und die Signaturkosten effizienter ist als alle bekannten Code-basierten Threshold Ring Signaturverfahren.

Contents

1 Introduction

Today, cryptography is embedded in all aspects of our life to protect our privacy. The security of many popular transactions such as: online shopping, secure electronic mail, automatic software updates, secure computer access, health care services, etc., can be ensured using cryptographic methods. Security is basically defined as integrity, confidentiality, authentication, and non-repudiation. To achieve these security services there are some traditional cryptography tools, such as encryption schemes, digital signatures, and identification protocols. Cryptography can be broadly classified into symmetric cryptography which uses a single key that both the sender and recipient know, and asymmetric cryptography (or public-key cryptography) that uses two different keys: a public key and a private key. The main advantage of public-key cryptography compared to symmetric cryptography is to remove the need for in-person meetings or trusted couriers to exchange secret keys.

Currently, public key cryptography has been dominated by two major families of cryptographic classes: primitives whose security is based on the assumption that factorisation of large integers is a hard problem, such as the Rivest-Shamir-Adleman (RSA) algorithm [71], and primitives, whose security is believed to be contingent on the difficulty of the discrete logarithm problem, such as the Digital Signature Algorithm (DSA) [54].

Quantum computation arises much interest in cryptography, since Peter Shor found a polynomial-time algorithm to solve the factoring and discrete logarithm problems using quantum computers [78]. Therefore, it naturally follows that quantum computers would render all widely used public key cryptosystems insecure. This is one of the principal reason to motivate the research of alternatives that can resist quantum attacks. Such alternative systems are called post-quantum cryptosystems. The most promising ones, at least for the moment, are based on codes, lattices, hash functions, and multivariate systems over finite fields.

Since the publication of McEliece's encryption scheme in 1978 [52], which was the first attempt to introduce error-correcting codes in cryptography, code-based cryptography has received much more attention in recent years. Many other proposals to build cryptosystems followed based on the hardness of the syndrome decoding problem which is now well studied and strongly believed to hold. The class of cryptographic schemes build on error-correcting codes, encompasses public key encryption schemes (e.g. [52, 57]), signature schemes (e.g. [26, 49]), identification schemes (e.g. [81, 88]), as well as hash functions and stream ciphers (e.g. [5, 38]). This thesis focuses more specifically on the code-based identification and signature schemes.

Identification schemes are very useful and fundamental tools in many applications

such as electronic fund transfer and online systems for preventing data access by invalid users. Such schemes are typical applications of zero-knowledge interactive proofs [43], which are two-party protocols allowing a party called a prover to convince another party called a verifier, that it knows some secret piece of information, without the verifier being able to learn anything about the secret value except for what is revealed by the prover itself. Zero-knowledge identification schemes are of particular interest because it is possible to convert them into secure signature schemes through the very famous Fiat-Shamir paradigm [32].

Besides the fact that designing code-based identification schemes offer security against quantum attacks, these schemes have other good features. Firstly, they are usually very fast and easy to implement compared to schemes based on number-theoretic problems as they use only matrix-vector multiplications. Secondly, their security are directly related to the syndrome decoding problem. Finally, the complexity of attacks against code-based identification schemes can be given in the expected number of binary operations and not only through asymptotic estimations, as in the case of lattice-based cryptosystems for example.

At Crypto 1993, Stern proposed an efficient code-based identification scheme using binary random codes, which is still today the reference in this area. This scheme is a multiple round zero-knowledge identification protocol, where each round is a three-pass interaction between the prover and the verifier with a soundness error of 2/3. Unfortunately its major weakness is the high communication complexity, this comes from having to repeat the protocol many times in order to reach a prescribed level of authentication. Another drawback is the large public key size, as for all code-based schemes. For 80-bit security level and a cheating probability of 2^{-16} (a weak authentication level according the norm ISO/IEC-9798-5), the public key size of Stern's scheme is around 15 KB, where the size of the communication complexity is more than 5 KB.

This thesis makes a step further by improving the Stern's scheme in terms of the communication complexity and public key size. We contribute further by extending one of the most important paradigm in cryptography, namely the Fiat-Shamir paradigm. Finally, we present an improved code-based signature scheme with additional properties.

Summary of results

Chapter 2 gives the background for the remainder of the thesis. It introduces the principal notions for codes, hard problems in coding theory, and an overview featuring the existing code-based signature schemes. Chapters 3 - 5 present our results. Finally, we resume and point out open problems and future fields of research in Chapter 6. In the following, we present a short summary of Chapters 3 - 5.

Code-based Identification Scheme (Chapter 3) This chapter is based on [19] presented in SAC 2010. Starting from Stern's scheme, we present an improved identification scheme based on the hardness of the syndrome decoding problem defined

over $\mathbb{F}_q$. Our scheme is a multiple round zero-knowledge identification protocol, where each round is a five-pass interaction between the prover and the verifier with a soundness error of 1/2 instead of 2/3 by Stern'one. Due to this fact we can reach any desired authentication level in fewer rounds, which has a positive impact on the communication complexity. Our scheme reduces this complexity by a factor of 25% compared to Stern's scheme. Moreover, it permits to obtain a public key of size of 4 KB, whereas that of Stern's scheme is 15 KB for 80-bit security and a cheating probability of 2^{-16}. Further, we provide a comparison of our scheme to other post-quantum identification schemes having similar features to ours. At the end of this chapter, implementation results will be provided confirming the advantage of our construction compared to Stern's scheme and its dual version proposed by Véron [88]. As far as we are aware, there is no code-based identification scheme with better performance than our scheme using random codes.

Extended Security Arguments for Signature Schemes (Chapter 4) This chapter is based on [29] presented in AfricaCrypt 2012. It is motivated by the previous chapter and other recent proposals in different areas [18, 80, 19, 73, 2, 76]. In all these works, a number of five-pass identification schemes have been presented providing better communication complexity compared to three-pass identification schemes in their corresponding area. Indeed, they fall outside the original framework if we want to transform them into existentially unforgeable signature schemes, since the original framework works only for canonical (three-pass) identification schemes. Therefore we enlarge the class of identification schemes to which the Fiat-Shamir transformation can be applied in order to obtain new efficient signature schemes. Furthermore, we provide a security proof of the resulting signatures in the random-oracle model following the work of Pointcheval and Stern presented at Eurocrypt 1996 [65]. To this end, we extend the well known forking lemma which is the main tool of the proof. As an application, we show explicitly in this chapter how to convert Stern's and our identification scheme to signature schemes, after that, we give the security arguments of such transformations. Finally, we show the running times for our **C** implementations of the obtained signature schemes. As a result, we obtain three very fast signature schemes, thus it is possible to sign and verify in the order of milliseconds. However, the signature sizes are typically about 19 KB for our scheme and 25 KB for Stern's and Véron's schemes for 80-bit security.

Improved Code-based Threshold Ring Signature Scheme (Chapter 5) This chapter is based on [17] presented in WAIFI 2012. In this chapter we present an improved code-based threshold ring signature scheme, which is fully anonymous and unforgeable based on a proof of knowledge in the random-oracle model. Our proposal is obtained through the application of the extended Fiat-Shamir transform to our five-pass identification scheme presented in chapter 3. Since this latter scheme has a low soundness error allowing a specified security to be reached in few rounds, this fact is used to achieve a threshold ring signature scheme with shorter signature length,

smaller public key size and signature cost compared to Aguilar et al.'s scheme, which is the most efficient threshold ring signature scheme based on coding theory. At the end of this chapter, we give both the theoretical comparison and the implementation results of the two schemes, in order to confirm the advantage of our proposal in terms of performance.

2 Coding Theory and Cryptography

In this chapter we give some basic mathematical definitions and principal tools related to coding theory that are prerequisite for the following chapters. We also introduce the basic cryptographic schemes that we need throughout this thesis. Finally, we present an overview featuring the main proposals for code-based signature schemes.

2.1 Introduction

The study of error-correcting codes and the associated mathematics is known as coding theory. It was introduced in the middle of the 20th century by Claude Shannon in a paper called "A Mathematical Theory of Communication" [77]. This discipline deals with the transmission of messages over noisy channels (WiFi network) or the storage of data to unreliable media (CD, hard disk). More specifically, if we suppose that an *encoder* converts a message into a *codeword* which is transmitted over noisy communication channels. The main goal is to find an efficient *decoder* which is able to correct possible added errors and convert the transmitted codeword back into the original message. The efficiency of a decoder depends on the used code.

In 1978, McEliece was the first one to introduce error-correcting codes in cryptography by presenting his code-based encryption scheme [52]. The general idea is to identify encryption with encoding, which consists of mapping the message to be sent to a codeword in the public code and then adding errors to it. The decoding step is identified with decryption, where the receiver is able to decode the transmitted message by using secret knowledge.

2.2 Coding Theory

We give in this section the basic definitions of coding theory. For a more extensive background we refer to [77]. We limit ourselves to linear codes over finite fields, which we capture in the following definition.

Definition 2.1 (Error-correcting code)**.** Linear codes are k-dimensional subspaces of an n-dimensional vector space over a finite field $\mathbb{F}_q$, where k and n are positive integers with $k < n$, and q a prime power. In short, linear codes with these parameters are often denoted $[n, k]$-codes.
In this thesis, we denote a $[n, k]$-code by $\mathcal{C}$. The elements of the set $\mathcal{C}$ are called *codewords*. The co-dimension r of a code $\mathcal{C}$ is defined as $r = n - k$.

Remark 2.2. If q equals 2 we speak of *binary* linear codes, otherwise we speak of *q-ary* linear codes.

Remark 2.3. The ratio $R = k/n$ is known as code rate and measures the information rate, i.e. the proportion of useful data transmitted in each codeword.

Definition 2.4 (Error-correcting capability)**.** Let ω be a positive integer. We say that a code $\mathcal{C}$ is able to correct ω errors if, for each codeword, it is possible to detect and correct any configuration of ω errors occurred during its transmission.

Any subvectorspace of a code $\mathcal{C}$ is said to be a subcode of $\mathcal{C}$, we can also define a notion of subfield subcode which is often used for constructing some special codes.

Definition 2.5 (Subfield subcodes)**.** Let m be a positive integer and $\mathcal{C}$ be a code defined over an extension field $\mathbb{F}_{q^m}$ of $\mathbb{F}_q$. A subflied subcode $\mathcal{C}'$ of $\mathcal{C}$ is the restriction of $\mathcal{C}$ to $\mathbb{F}_q$:

$$\mathcal{C}' = \mathcal{C}_{|\mathbb{F}_q} = \mathcal{C} \cap \mathbb{F}_q^n.$$

As codes are treated as vector spaces, we often define them by the matrices related to the code.

Definition 2.6 (Generator matrix, parity-check matrix)**.** Let $\mathcal{C}$ be a linear code over $\mathbb{F}_q$. A generator matrix G of $\mathcal{C}$ is an $k \times n$ matrix whose rows form a basis of $\mathcal{C}$:

$$\mathcal{C} = \{xG : x \in \mathbb{F}_q^k\}.$$

A parity-check matrix H of $\mathcal{C}$ is an $r \times n$ matrix whose rows form a basis of the orthogonal complement of the vector subspace $\mathcal{C}$, i.e. it holds that,

$$\mathcal{C} = \{x \in \mathbb{F}_q^n : Hx^T = 0\}.$$

Definition 2.7 (Syndrome)**.** The syndrome of a vector $x \in \mathbb{F}_q^n$ with respect to H is the column vector $Hx^T \in \mathbb{F}_q^r$.

In particular, the elements of a code $\mathcal{C}$ have zero as syndromes.

Remark 2.8. The two matrices G and H are not unique and by a linear transformation one can easily determine a parity-check matrix from a generator matrix for a linear code $\mathcal{C}$. In particular, given a matrix G in systematic form, i.e., $G = (I_k|R)$ where I_k denotes the $k \times k$ identity matrix and $R \in \mathbb{F}_q^{k\times(n-k)}$, then $H = (-R^T|I_{n-k})$ is a parity-check matrix for the code $\mathcal{C}$.

Definition 2.9 (Hamming distance, Hamming weight, minimum distance)**.** The Hamming distance $\mathrm{dist}(x, y)$ between two words $x, y \in \mathbb{F}_q^n$ is the number of coordinates where they differ. The Hamming weight of a vector x is the number of non-zero entries. We use $\mathsf{wt}(x)$ to represent the Hamming weight of x. The minimum distance d of a linear code $\mathcal{C}$ is the smallest Hamming distance between different codewords.

In many cases, we simply write weight instead of Hamming weight and distance instead of Hamming distance.

Remark 2.10. The minimum distance of a code $\mathcal{C}$ is fundamental to determine its error-correcting capability. More formally, for a word x and $\omega \leq \left\lfloor \frac{(d-1)}{2} \right\rfloor$, a decoding algorithm uniquely outputs the closet codeword c if d is the minimum distance of $\mathcal{C}$. In particular, a linear code with minimum distance d has an error-correcting capability $\omega = \left\lfloor \frac{(d-1)}{2} \right\rfloor$. A code with these properties is denoted an $[n, k, \omega]$-code.

In the following sections we introduce some special types of codes that we need in some parts of this thesis.

2.2.1 Goppa codes

Goppa codes were introduced by Valery D. Goppa in [85, 86]. This class of codes was used by McEliece to define his cryptosytem. We first define generalized Reed-Solomon codes which are strongly related to the class of Goppa codes.

Definition 2.11 (Generalized Reed-Solomon code)**.** Let q a prime power and m be a positive integer. Given a sequence $L = (L_0, \ldots, L_{n-1}) \in \mathbb{F}_{q^m}^n$ such that the L_i are pairwise different elements of $\mathbb{F}_{q^m}$ and a sequence $D = (D_0, \ldots, D_{n-1})$ where D_i are nonzero elements of $\mathbb{F}_{q^m}$. The generalized Reed-Solomon code $GRS_\omega(L, D)$ is the $[n, k, \omega]$-code defined by the following parity-check matrix

$$H = \begin{bmatrix} 1 & \ldots & 1 \\ L_0 & \ldots & L_{n-1} \\ \vdots & & \vdots \\ L_0^{r-1} & \ldots & L_{n-1}^{r-1} \end{bmatrix} \cdot \begin{bmatrix} D_0 & & 0 \\ & \ddots & \\ 0 & & D_{n-1} \end{bmatrix}.$$

The alternant code; denoted $\mathscr{A}(L, D)$, is a subfield subcode of the generalized Reed-Solomon code $GRS_\omega(L, D)$.

Definition 2.12 (Goppa codes)**.** Given a sequence $L = (L_0, \ldots, L_{n-1})$ such that the L_i are pairwise different elements of $\mathbb{F}_{q^m}$ and a polynomial $g(x) \in \mathbb{F}_{q^m}[x]$ of degree ω such that $g(L_i) \neq 0$ for $0 \leqslant i < n$. The Goppa code $\Gamma(L, g)$ over $\mathbb{F}_q$ is $\mathscr{A}(L, D)$, the alternant code over $\mathbb{F}_q$ that corresponds to $GRS_\omega(L, D)$, where $D = (g(L_0)^{-1}, \ldots, g(L_{n-1})^{-1})$.

2.2.2 Quasi-cyclic, quasi-dyadic codes

Code-based cryptosystems suffer from a major drawback, they require a very large public key which makes them very difficult to use in many practical situations. To mitigate this drawback, there have been some proposals which consists in replacing a matrix defining a code by a particular type of matrices with a very compact representation. We present two examples of such proposals which are relevant to this thesis. The first one by Berger et al. consists in using quasi-cyclic codes [11].

The second one by Misoczki and Barreto uses quasi-dyadic codes [53]. A parity-check matrix H is called a quasi-cyclic matrix (resp. quasi-dyadic matrix), if H is a $r_0 b = r \times n = n_0 b$ matrix consisting of $r_0 \times n_0$ blocks of $b \times b$ spare circulant (resp. dyadic) submatrices, for some integer b. The same holds true for the case of a generator matrix by replacing $r = r_0 b$ and $k = k_0 b$.

A circulant matrix is defined by a vector $(a_1, a_2, \ldots, a_b) \in \mathbb{F}_q^b$ and has the following form:

$$R^* = \begin{pmatrix} a_1 & a_2 & a_3 & \ldots & a_b \\ a_b & a_1 & a_2 & \ldots & a_{b-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_2 & a_3 & a_4 & \ldots & a_1 \end{pmatrix}.$$

A matrix $(I_r | R^*)$ is called double circulant matrix.

A dyadic matrix is recursively defined: any 1×1 matrix is dyadic and for $p > 1$, a $2^p \times 2^p$ dyadic matrix has the form:

$$R^* = \begin{pmatrix} B & C \\ C & B \end{pmatrix},$$

where B and C are $2^{p-1} \times 2^{p-1}$ dyadic matrices. To give an example, an 4×4 dyadic matrix has the following form:

$$R^* = \begin{pmatrix} a & b & c & d \\ b & a & d & c \\ c & d & a & b \\ d & c & b & a \end{pmatrix},$$

where $a, b, c, d \in \mathbb{F}_q$.

The advantage of a parity-check quasi-cyclic resp. quasi dyadic matrix is the fact that the whole matrix can be reconstructed from the knowledge of the first rows of the $r_0 k_0$ blocks. This is the trick to reduce a public key element.

2.3 Provable Security

The concept of provable security in cryptography has the goal to offer formally veriable guarantees that no adversarial strategy will be successful to break a scheme as long as certain assumptions hold. Most of all, it is possible to provide a proof of security using a black-box polynomial-time Turing reductions. A reduction proof works by the method of contradiction; if there exists an adversary that breaks the scheme, then there exists an algorithm that breaks the underlying problem. There are mainly two models of providing this security proof, the first one is called the

standard model and the second one is called the random-oracle model. These two models will be defined below.

Definition 2.13 (Standard model)**.** A standard model in cryptography is the model of computation in which the adversary is only limited by the amount of time and computational power available.

This is the "real" world scenario. Schemes that are proven secure using only complexity assumptions are said to be secure in the standard model.

Most of all, it is difficult to achieve security proofs in the standard model, therefore, the idea is to replace cryptographic primitives by idealized versions, called random oracles.

Definition 2.14 (Random oracle)**.** A random oracle is a mathematical abstraction that works as a theoretical black box, that is, an oracle that answers to every query with a uniformly random sample. For any specific query, the output returned is always the same.

Random oracles are very useful to represent functions that need to have a truly random behavior, most commonly cryptographic hash functions. When used in reduction proofs, the random oracle allows the reduction algorithm to adaptively program the input-output behavior outside of the view of the remaining algorithms. This technique allows security proofs for schemes that are otherwise hard or impossible to prove secure under standard assumptions.

Definition 2.15 (Random-oracle model)**.** A Random-oracle model is a heuristic used to provide security arguments for cryptographic protocols by modeling cryptographic hash functions with perfectly random functions.

When we give a security proof for a scheme in the random oracle model, we say that this scheme is secure in the random-oracle model.
One major application of random oracles is the Fiat-Shamir heuristic, which allows to turn interactive identication protocols into digital signature schemes.

2.4 Cryptographic Primitives

We begin by introducing some notations and briefly reviewing some definitions. A function $\mu(\cdot)$ is negligible in n (a positive integer), or just negligible, if for every positive polynomial $p(\cdot)$ and all sufficiently large n, it holds that $\mu(n) < 1/p(n)$. Otherwise, we call $\mu(\cdot)$ non-negligible. Note that the sum of two negligible functions (resp. non-negligible) is again negligible (resp. non-negligible) whereas the sum of one non-negligible function $\pi(\cdot)$ and one negligible function $\mu(\cdot)$ is non-negligible, i.e. there exists a positive polynomial $p(\cdot)$ such that for infinitely many n's it holds that $\pi(n) + \mu(n) > 1/p(n)$.

Two distributions ensembles $\{X_n\}_{n\in\mathbb{N}}$ and $\{Y_n\}_{n\in\mathbb{N}}$ are said to be (computationally) indistinguishable, if for every non-uniform polynomial-time algorithm D, there exists a negligible function $\mu(\cdot)$ such that

$$|\Pr[\,D(X_n)=1]-\Pr[\,D(Y_n)=1]\,|\leq\mu(n).$$

A random variable X has min-entropy k, denoted $H_\infty(X)=k$, if

$$\max_x \Pr[X=x]=2^{-k}.$$

Here, we recap the definitions and security models of hash functions.

Definition 2.16 (Hash function)**.** Let $\mathcal{H}$ be a function on A whose range B is a set of strings of fixed length n. Then $\mathcal{H}$ is a cryptographic hash function if it satisfies the following properties:

- Computability: For all $x\in A$ it is easy to compute $\mathcal{H}(x)$.
- Preimage resistance: For all $y\in B$ it is computationally infeasible to find $x\in A$ such that $y=\mathcal{H}(x)$.
- Second-preimage resistance For all $x\in A$ it is computationally infeasible to find $x'\neq x$ such that $\mathcal{H}(x')=\mathcal{H}(x)$.
- Collision resistance: It is computationally infeasible to find $x,x'\in A$ such that $x\neq x'$ and $\mathcal{H}(x)=\mathcal{H}(x')$.

The value $\mathcal{H}(x)$ is called message digest or simply digest. Clearly, all the properties are required in order to ensure that a malicious adversary is unable to modify the input without changing its digest. Usually the data is encoded in binary, and we have $A=\{0,1\}^*$ (bit-strings of arbitrary length) and $B=\{0,1\}^n$.

In the following we recap the definitions and security models for identification and signature schemes.

Definition 2.17 (Identification scheme)**.** An identification scheme consists of a probabilistic polynomial-time algorithm KeyGen and two probabilistic polynomial-time interactive algorithms $\mathcal{P}$ and $\mathcal{V}$ with the following properties:

- The algorithm KeyGen is a key generation algorithm. It takes as input a security parameter κ and outputs a pair of strings $(\mathsf{sk},\mathsf{pk})$, pk is called a public key, and sk is called a secret key.
- $\mathcal{P}$ receives as input the pair $(\mathsf{sk},\mathsf{pk})$ and $\mathcal{V}$ receives as input pk. After an interactive execution of $\mathcal{P}$ and $\mathcal{V}$, $\mathcal{V}$ outputs an 1 (indicating "accept") or a 0 (indicating "reject"). For a given sk and pk, the output of $\mathcal{V}$ at the end of this interaction is a probability space and is denoted by $<\mathcal{P}(\mathsf{sk},\mathsf{pk}),\mathcal{V}(\mathsf{pk})>$.

For the security proof we use the concept of zero-knowledge interactive proof of knowledge system. In such context, $\mathcal{P}$ has as goal to convince a $\mathcal{V}$ that a given string x belongs to a language L, without revealing any other information. This kind of proof satisfies three properties:

1. Completeness: any true theorem can be proven. That is, $\forall x \in L$ $(< \mathcal{P}(\mathsf{sk}, \mathsf{pk}), \mathcal{V}(\mathsf{pk}) > [x] = 1) \geq 1 - \mu(n)$. ($\mu(n)$ is a negligible function on some security parameter κ).

2. Soundness: no false theorem can be proven. That is, $\forall x \notin L$ $\forall \mathcal{P}'$ (malicious) $(< \mathcal{P}'(\mathsf{sk}, \mathsf{pk}), \mathcal{V}(\mathsf{pk}) > [x] = 1) \leq 1/2$, where $\mathcal{P}'$ denotes any entity playing the role of the real $\mathcal{P}$.

3. Zero-Knowledge: anything one could learn by listening to $\mathcal{P}$, one could also have simulated by oneself. That means, there exists a probabilistic polynomial-time simulator (Sim) such that no polynomial-time distinguisher can tell whether $\mathcal{V}$ is interacting with an honest prover or interacting with the simulator.

Remark 2.18. Assuming that one round of an identification protocol has a soundness error equals λ. In order to detect cheating $\mathcal{P}$, the protocol has to be run several times. For example, to achieve an authentication level L, one determine the minimum number of rounds δ such that: $\lambda^{\delta} \leq L$.

Definition 2.19 (Signature scheme)**.** A digital signature scheme is a collection of the following algorithms $\mathrm{S} = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$ defined as follows.

$\mathsf{KGen}(1^{\kappa})$ is a probabilistic algorithm which, on input a security parameter κ, outputs a secret and a public key $(\mathsf{sk}, \mathsf{pk})$.

$\mathsf{Sign}(\mathsf{sk}, M)$ is a probabilistic algorithm which, on input a secret key sk, a message M, outputs a signature σ.

$\mathsf{Vf}(\mathsf{pk}, M, \sigma)$ is a deterministic algorithm which, on input of a public key pk, a message M and a signature σ, outputs either 1 (= valid) or 0 (= invalid).

We require correctness of the verification, i.e., the verifier always accept genuine signatures. More formally, for all $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KGen}$, any message M, any $\sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, M)$, we always have $\mathsf{Vf}(\mathsf{pk}, M, \sigma) = 1$.

For signature schemes we require that no outsider should be able to forge a signer's signature. The following definition captures this property formally.

Definition 2.20 (Unforgeability of a signature scheme)**.** *A signature scheme* $\mathrm{S} = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$ *is unforgeable under adaptive chosen message attacks if for any efficient algorithm* $\mathcal{A}$ *making at most* q_s *oracle queries, the probability that the following experiment returns* 1 *is negligible:*

Experiment $\mathsf{Unforgeability}^{\mathrm{S}}_{\mathcal{A}}(\kappa)$
$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KGen}(1^{\kappa})$
$(\sigma^*, M^*) \leftarrow \mathcal{A}^{O\mathsf{Sign}(\mathsf{sk},\cdot)}(\mathsf{pk})$
$O\mathsf{Sign}(\cdot)$ *on input* M
outputs $\sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, M)$
Return 1 *iff*
$\mathsf{Vf}(\mathsf{pk}, M^*, \sigma^*) = 1$ *and*
M^* *was not queried to* $O\mathsf{Sign}(\mathsf{sk}, \cdot)$ *by* $\mathcal{A}$

The probability is taken over all coin tosses of KGen, Sign, *and* $\mathcal{A}$.

Definition 2.20 captures unforgeability against adaptively chosen message attacks for signature schemes. Unforgeability against no-message attacks is defined analogously but q_s must be 0.

2.5 Code-based Cryptography

We describe in the following the main hard problems on which the security of code-based schemes presented in this paper relies. We denote by $x \xleftarrow{\$} A$ the uniform random choice of x among the elements of a set A. The symbol | denotes the concatenating operator.

2.5.1 Hard problems

Definition 2.21 (Binary Syndrome Decoding Problem (SD))**.**
Input : $H \xleftarrow{\$} \mathbb{F}_2^{r\times n}$, $y \xleftarrow{\$} \mathbb{F}_2^r$, and an integer $\omega > 0$.
Find : a word $s \in \mathbb{F}_2^n$ such that $\mathsf{wt}(s) \leq \omega$ and $Hs^T = y$.

This problem was proven to be NP-hard in 1978 [10]. A dual version of the previous problem, using the generator matrix G instead of the parity-check matrix H of the code $\mathcal{C}$, can be defined as follows.

Definition 2.22 (General Decoding Problem (G-SD))**.**
Input : $G \xleftarrow{\$} \mathbb{F}_2^{k\times n}$, $y \xleftarrow{\$} \mathbb{F}_2^n$, and an integer $\omega > 0$.
Find : A pair $(x, e) \in \mathbb{F}_q^k \times \mathbb{F}_q^n$, where $\mathsf{wt}(e) \leq \omega$ s.t $xG + e = y$.

GD states that given a vector $y \in \mathbb{F}_2^n$, find the (unique) codeword $x \in \mathcal{C}$, such that $\mathsf{wt}(x - y)$ is minimal. GD is also proven to be NP-hard. Moreover, it is assumed that it is hard not only for some worst-case instances, but hard on average.

An extension of the binary syndrome decoding (SD) problem over an arbitrary finite field can be formulated as well.

Definition 2.23 (q-ary Syndrome Decoding (qSD) problem)**.**
Input : $H \xleftarrow{\$} \mathbb{F}_q^{r\times n}$, $y \xleftarrow{\$} \mathbb{F}_q^r$, and an integer $\omega > 0$.
Find : a word $s \in \mathbb{F}_q^n$ such that $\mathsf{wt}(s) \leq \omega$ and $Hs^T = y$.

The qSD problem was proven to be NP-hard by A. Barg in 1994 [6]. We define a variant of this problem in the case $y = 0$.

Definition 2.24 (q-ary Minimum Distance (qMD) problem)**.**
Input : $H \xleftarrow{\$} \mathbb{F}_q^{r\times n}$, and an integer $\omega > 0$.
Find : a word $s \in \mathbb{F}_q^n$ such that $\mathsf{wt}(s) \leq \omega$ and $Hs^T = 0$.

Notably the difficulties of solving the two problems (qSD and qMD) are equivalent [83]. The intractable assumptions associated to these problems are respectively denoted by qSD assumption and qMD assumption.

We now present a very important bound for linear codes. Gilbert [40] and Varshamov [87] independently developed bounds on the maximum size of a code. Based on these bounds, Barg [6] proposed the related Gilbert-Varshamov distance.

Definition 2.25 (Gilbert-Varshamov distance, Gilbert-Varshamov bound)**.** Let $\mathcal{C}$ be an $[n, k]$ linear code over $\mathbb{F}_q$, the Gilbert-Varshamov (GV) distance is defined as the maximum integer d_0 such that

$$\sum_{j=0}^{d_0-1} \binom{n}{j}(q-1)^j \leq q^{n-k} \ .$$

If a weight ω of $\mathcal{C}$ satisfies $\omega \leq d_0$, we have a unique solution to SD problem. Otherwise, multiple solutions exist [62]. It follows that decoding problems are meaningful only if the weight ω is small.

Let $H_q(x)$ be the q-ary entropy function, given by:

$$H_q(x) = x\log_q(q-1) - x\log_q(x) - (1-x)\log_q(1-x) \ .$$

It is well-known that for sufficiently large n random linear codes reach the so called the Gilbert-Varshamov (GV) bound:

Suppose $0 \leq \xi \leq (q-1)/q$. Then there exists an infinite sequence of (n, k) q-ary linear codes with $d/n = \xi$ (d is the minimal distance of the code) and rate $R = k/n$ satisfying the inequality:

$$R \geq 1 - H_q(\xi) \qquad \forall n.$$

Remark 2.26. Like random binary codes, it is proved in [36] that quasi-cyclic codes can asymptotically approach the Varshamov-Gilbert bound. This issue is still an open problem for quasi-dyadic codes.

2.5.2 Attacks on code-based cryptosystems

The most successful attacks on code-based cryptosystems can be classified in two major classes: decoding attacks and structural attacks. The first class is used to decode a given cipher which has no visible structure, and the second one try to recover the structure of the secret code from the public key. In this thesis, we consider only decoding attacks, since we use only random codes in our constructions.

The most efficient decoding attack in this case is the Information Set Decoding (ISD) algorithm. Some improvement of this algorithm have been developed by Peters [63], Niebuhr et al. [56], Bernstein et al. [12], and recently by Becker et al. in [9] and by May et al. [50].

Given a $r \times n$ parity-check H of a code, and $Hs^T = y$, where s is a vector of weight ω and y its syndrome. The ISD algorithms take as input the vector y and try to recover s. To do this, the main idea consists in applying a permutation matrix P to H in the hope that all columns corresponding to error positions in s are moved to the left side of the matrix H. After using the Gaussian elimination, one gets a matrix $H' = [I_r|R]$ (R is a $r \times n - r$ matrix) and the row operations are performed on y in order to get a vector y'. If y' has a weight smaller or equal than ω, the ISD algorithm succeds, otherwise one restarts this algorithm again.

The ISD algorithm is often used as a tool to determine the parameters $n, k = n-r$ and ω of a given code required to achieve the desired security level (e.g. 80-bit or 128-bit security) and we denote the workfactor of this algorithm by $\mathrm{WF}_{\mathrm{ISD}}$.

2.5.3 Code-based signature schemes

Signature schemes are among the most useful and recurring cryptographic schemes. Usually, there are three main approaches for constructing code-based signature schemes. The first one requires a trapdoor like RSA signature scheme. The second one uses a well-known Fiat-Shamir paradigm converting zero-knowledge identification protocols into secure signature schemes. The last approach is generic, it is similar to El Gamal signature and consists of defining particular sets where the signer is able to decode any syndrome.

In this section we present an overview featuring the main proposals following the first and the third approaches. The second approach is addressed more explicitly in Chapter 3 of this thesis.

Courtois, Finiasz and Sendrier (CFS) signature scheme. The CFS scheme was introduced by Courtois, Finiasz and Sendrier in Asiacrypt 2001 [26]. This scheme is not quite as successful as the RSA signature, due to the large signing time. This is due to the fact that decoding any random element into codeword is not guaranteed. The authors of the CFS scheme uses Goppa codes, which have a good proportion of decodable words and choose parameters such that this proportion is reasonable. For a Goppa code of length $n = 2^m$ (m positive integer) and ω as correcting capability, the number of decoding attempts to get one signature is in average approximately around $\omega!$. The authors of [26] suggested the following parameters $(m, \omega) = (16, 9)$ for a 80-bit security. For such parameters the signature length is 144 bits, however, in order to sign, it is necessary to repeat the algorithm in average 9! times. An additional disadvantage is a public key size which can attain 1152 KB.

This scheme can be described as follows, given a message M to be signed and a secure hash function $\mathcal{H}$. The idea of the CFS algorithm is to compute $M_i = \mathcal{H}(\mathcal{H}(M)|i)$ starting for i by 0 and increasing at each try until M_i is decodable. This

syndrome M_i is decoded into a word s of length n using the decoding algorithm, such that $Hs^T = M_{i_0}$, where i_0 is the smallest value of i for which a decoding is possible. The signature consists of $\{s, i_0\}$. Algorithm 2.1 describes more explicitly the CFS signature scheme.

Algorithm 2.1 CFS algorithm

Parameters: $H \in \mathbb{F}_2^{r \times n}$: parity-check matrix of Goppa code, $\mathcal{H}$ a collision resistant hash function.

▷ Signature:

1: Hash the message M (to be signed) into $\mathcal{H}(M)$
2: Compute $M_i = \mathcal{H}\left(\mathcal{H}(M)|i\right)$ for $i = 0, 1, 2 \cdots$
3: Find i_0 the smallest value of i such that M_i decodable
4: Using the decoding algorithm to compute s such that $Hs^T = M_{i_0}$
5: Signature of M: $\{s, i_0\}$

▷ Verification:

6: Compute $b_1 = Hs^T$
7: Compute $b_2 = \mathcal{H}(\mathcal{H}(M)|i_0)$
8: Compare b_1 and b_2, if they are equal the signature is valid

In view of Bleichenbacher's attack described in [34], the preliminary parameters of the CFS had to be increased. The authors of [34] suggested the parameters must be set to $(m, \omega) = (15, 12)$ or $(16, 10)$. This leads to an increase of the public key size or of the signature cost by an exponential factor in parameters $\{m, \omega\}$.

Finiasz suggested in [33] a way to increase the security of the CFS while keeping the parameters as small as possible. The idea of his proposal consists in performing a parallel decoding to generate two CFS signatures using two different hash-functions for the same message. In this case, an attacker has to produce two forgeries for one message, this makes the decoding attack much harder compared with the regular CFS. The same idea can be generalized to several parallel decodings. However, the gain in security offered by this proposal is at the cost of a huge increase in signing/verification cost and in the signature size, due to the complete decoding requirement.

Motivated by the drawback that the CFS scheme requires a large memory requirement, Barreto et al. [7] proposed an improved version of CFS using QD Goppa codes instead of the standard Goppa codes. This modification allows to reduce the key size by a factor of 4 in practice and to speed-up the computation by using the QD structure. However, this improvement has a disadvantage of increasing the number of signing attempts by a factor of 2.

The security proof of the CFS scheme was presented in 2007 by Dallot [27] using a reduction to the hardness of syndrome decoding and code indistinguishability. However, in 2010, Faugère et al. showed in [31] that Goppa codes of very high rate $R = k/n$ can be distinguished from random codes. This leads to the invalidity of the dallot's security proof. Recently, the authors in [70] addressed this problem and showed the existential unforgeability of CFS signature scheme under chosen message

attacks.

Kabatianskii, Krouk, and Smeets (KKS) signature scheme. KKS signature scheme was proposed by Kabatianskii, Krouk and Smeets in 1997 [49], this scheme is based on random linear codes. Unlike the CFS signature scheme, the KKS scheme avoids the need of using a decoding algorithm. The main idea of this construction consists in building a public linear matrix for which the signer is able to decode any syndrome obtained by column combination of this matrix. The security of the KKS scheme has been investigated by Cayrel et al. in [21]. The authors of [21] showed that a passive attacker intercepting just a few message/signature pairs can efficiently find the private key. They gave precisely the number of signatures (at most 20 signatures) required to achieve this target. Furtheremore, they broke all parameters proposed in [49] and suggested new secure ones instead. Therefore, the KKS signature scheme can be used only as one-time signature. For secure parameters, the KKS signature scheme provide in average a reasonable signature size (0.3 KB), unfortunately the public key size is large (25 KB). Recently, Otmani and Tillich showed in [60] an attack that break all proposed parameters for the KKS scheme without even needing to know a single message/signature pair. We should mention that this attack doesn't break the scheme itself.

Barreto, Misoczki, and Simplcio (BMS) signature scheme. In 2010, Barreto et al. proposed a variant of the KKS signature scheme [8], the aim of this work was to have a one-time signature scheme with a security proof. This is accomplished by combining the idea of Schnorr [74] and KKS [49]. More explicitly, this construction modifies the KKS scheme by introducing the use of a hash function and adding an error vector to the signature. The authors proved that BMS scheme is one-time existential unforgeability against chosen-message attacks based on the hardness of decoding random binary codes. The proposed parameters for this scheme are also affected by the attack by Otmani and Tillich.

Gaborit, Schrek (GS) signature scheme. Recently, Gaborit and Schrek proposed a code-based one-time signature scheme (GS) [37] based on some special codes with an automorphism group in order to decrease the public key size for the KKS scheme. The idea of GS construction consists of using one given syndrome to construct several decodable syndromes by using some combinatorial properties. This scheme can be considered as a trade-off between the size of the signature and the size of the public key. The authors obtained public key sizes less than 2.25 KB and a signature length of 0.8 KB.

Table 2.1 shows a comparison of all previous presented signature schemes for 80-bit security without taking into account the recent attack by Otmani and Tillich.

	CFS	**KKS**	**BMS**	**GS**
Public key size (KB)	720	25	113.5	2.25
Signature length (KB)	0.02	0.3	0.45	0.8
Approach	trapdoor	generic	generic	generic

Table 2.1: Comparison of code-based signature schemes.

Code-based signature schemes with special properites

The development of cryptosystems with additional properties is one of the recent hot research topics. Based on coding theory, there exist only few constructions of such schemes. The existing schemes are: ring signature [91], blind signature [61], identity-based identification and signature schemes [22, 8], and threshold ring signatures [3, 4, 17]. We refer the readers to [16] for a general overview on this topic.

3 Code-based Identification Schemes

Public-key identification schemes are typically applied in cryptography in order to reach one of the main objectives, namely access control. These schemes enable one party to authenticate to another via an insecure channel without disclosing any additional information that might be used by an impersonator. The most efficient code-based identification scheme was proposed by Stern in Crypto 1993. In this chapter we propose an improved variant of Stern's scheme by designing a five-pass identification scheme reducing the soundness error from 2/3 to 1/2. Due to this fact we can reach any desired level of security in fewer rounds, which permits to reduce the communication complexity for our scheme by a factor of 25% compared to Stern's one, typically for a cheating probability of 2^{-16}. Moreover, our scheme permits to obtain a public key of size of 4 KB, whereas that of Stern's scheme is 15 KB for the same level of security. Another advantage of our scheme is that, its security is directly based on the hardness of the syndrome decoding problem defined over $\mathbb{F}_q$. Overall our scheme has the good features of satisfying a zero-knowledge security proof, a small communication complexity, and a small public key size compared to all previous code-based identification schemes using random codes.

This chapter is based on a joint work with Pierre-Louis Cayrel and Pascal Véron [19]. It was presented at the 17th Annual International Workshop on Selected Areas in Cryptography (SAC 2010) in Waterloo, Ontario, Canada.

3.1 Introduction.

An identification (ID) scheme is a series of messages exchanged between two entities, called prover and verifier, in order to enable a prover to convince a verifier that it knows a given secret key corresponding to a public key assumed to be held by the verifier. The minimum security of such schemes should be that a passive observer who sees the interaction should not be able to perform his own interaction and successfully impersonates the prover. A formal definition of an ID scheme and its security properties are given in Section 2.4.

Since the introduction of the famous Fiat-Shamir's scheme [32], there have been many other proposals (e.g. [76, 45, 79, 59, 41]) for constructing secure ID schemes. Most of them are based on problems from number theory. Such proposals require fairly costly multiplication and exponentiation operations. Another potential problem is that the security of those schemes are based on problems which can be solved in polynomial time if (or when) quantum computers become reality.

After introducing the first encryption scheme by McEliece in 1978 using error-correcting codes [52], there have been many attempts to build secure code-based ID

schemes. The first scheme was proposed in 1989 by Harari in [46], unfortunately it has been proved to be insecure in [89]. A second scheme was proposed by Stern in 1989 [84], but this proposal was inefficient. Another scheme proposed by Girault in 1990 [42] has been demonstrated insecure in [75]. Eventually, the first efficient and secure code-based ID scheme was proposed at Crypto 1993 by Stern [81], which is still the reference in this area. Stern's scheme is a multiple-round zero-knowledge protocol, where each round is a three-pass interaction between the prover and the verifier with a soundness error of 2/3. A few years later, Véron designed in [88] a scheme which slightly decreases the communication complexity but at the same time it increases the size of the public key compared to Stern's one.

Stern's and Véron's ID schemes are very interesting since their security is directly related to a hard problem. They are very fast and usually easy to implement. Moreover, they can be turned into signature schemes through the Fiat-Shamir's paradigm [32]. Meanwhile, there are two strong drawbacks:

1. Since the soundness error is 2/3 for these two constructions instead of 1/2 as in the case of Fiat-Shamir's ID protocol based on integer factorization [32], they use more rounds to achieve the same security, typically 28 rounds for a cheating probability of 2^{-16}.

2. The public key size is very large, typically 15 KB.

In this chapter, we propose an improvement of the Stern's and Véron's ID schemes by defining a five-pass code-based ID scheme for which the success probability of a cheater is 1/2 and where the public key size is reduced as well. We reach this without losing provable security, since our proposal uses random codes over $\mathbb{F}_q$ which is proved to be NP-hard [6].

The content of this chapter is organized as follows. We present a short description of Stern's and Véron's ID schemes in Section 3.2. Afterwards, we give in Section 3.3 a detailed description of our proposal, we discuss the security, and we show the advantage of our construction by giving a theoretical comparison to Stern's, Véron's schemes and to other post-quantum ID schemes having some features similar to our scheme. Finally, we show in Section 3.4 a performance aspect of our construction by providing implementation results.

3.2 Stern's and Véron's Schemes

In the following we briefly describe the Stern's and Véron's schemes and their properties.

Stern's scheme

Stern's ID scheme uses a fixed binary $r \times n$ matrix H which is common to all provers. If H is chosen randomly, it provides a parity check matrix for a code $[n, k, \omega]$ ($r =$

$n - k$) with asymptotically good minimum distance given by the (binary) Gilbert-Varshamov (GV) bound. The prover's private key s is a binary vector of length n and small weight $\mathsf{wt}(s) = \omega$ (e.g. $\omega \approx$ GV bound), which corresponds to the syndrome $Hs^T = y$, the public key. In one round of the Stern's protocol, the prover identifies itself by proving his knowledge of the vector s without revealing any information on it. To this end, two blending factors are used: a permutation and a random vector. However, a dishonest prover not knowing s can have success with probability up to 2/3 in any given round. Thus, the protocol has to be run several times to detect cheating provers. The number of rounds depends on the authentication level needed. For instance to achieve the weak and strong authentication levels of 2^{-16} and 2^{-32} according the norm ISO/IEC-9798-5, one needs respectively 28 and 56 rounds. The security of this scheme relies on the hardness of the binary syndrome decoding (SD) problem, that is on the difficulty of determining the preimage s of $y = Hs^T$.
The Stern's scheme has two parts: a key generation algorithm, shown in Figure 3.1, and an ID protocol as given in Figure 3.2.

In what follows, $\mathcal{H}$ denotes a hash function and S_n the symmetric group of degree n.

KeyGen:
Let κ be the security parameter
Choose n, r, ω, such that $\mathrm{WF}_{\mathrm{ISD}}(n, r, \omega, 2) \geq 2^{\kappa}$
$H \xleftarrow{\$} \mathbb{F}_2^{r \times n}$
$s \xleftarrow{\$} \mathbb{F}_2^n$, s.t. $\mathsf{wt}(s) = \omega$.
$y \leftarrow Hs^T$
Output $(\mathsf{sk}, \mathsf{pk}) = (s, (y, H, \omega))$

Figure 3.1: Stern key generation algorithm.

Véron's scheme

In 1994, Véron proposed a dual construction of Stern's ID scheme [88]. He used a $k \times n$ public generator matrix G defining a random linear binary code and designed an ID scheme based on the hardness of the G-SD problem. For this scheme, a prover demonstrates his knowledge of a pair (e, x) such that $xG \oplus e = y$ (public), where e is a binary vector of weight ω and length n and x is a random binary vector of length k. Véron's scheme decreases slightly the communication complexity but it increases the size of public key compared to Stern's scheme. The secret key is protected by using the same two techniques as in the Stern's construction: the transformation by means of a permutation, and the addition of a random vector. Véron's scheme is also a multiple-rounds ID protocol, where each round is a three-pass interaction with a soundness error of 2/3 to succeed in the protocol without the knowledge of

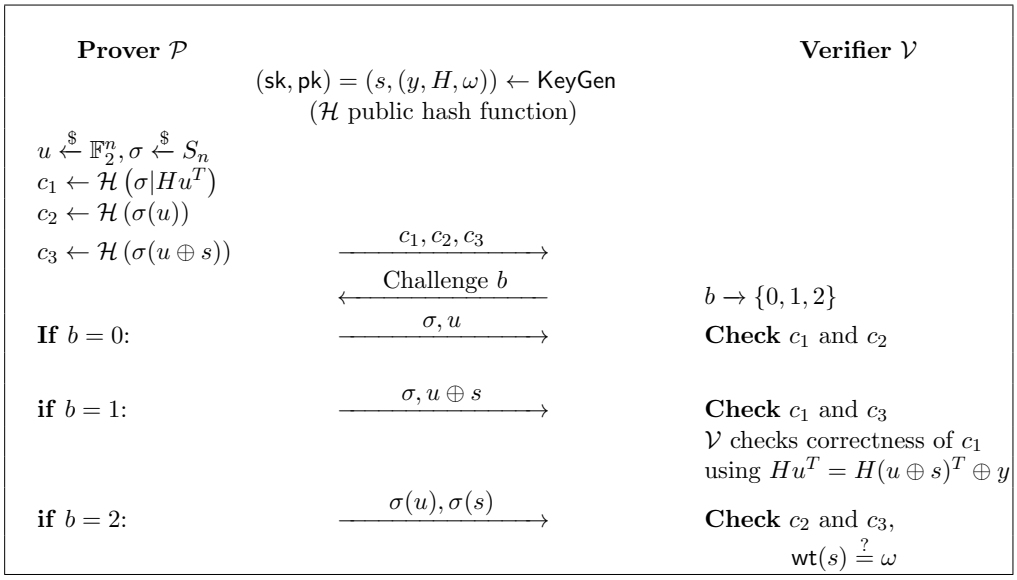

Figure 3.2: Stern identification protocol.

KeyGen:
Let κ be the security parameter
Choose n, k, and ω such that $\mathrm{WF}_{\mathrm{ISD}}(n, k, \omega, 2) \geq 2^\kappa$
$G \xleftarrow{\$} \mathbb{F}_2^{k \times n}$
$(x, e) \xleftarrow{\$} \mathbb{F}_2^k \times \mathbb{F}_2^n$, s.t. $\mathsf{wt}(e) = \omega$
$y \leftarrow xG \oplus e$
Output $(\mathsf{sk}, \mathsf{pk}) = ((x, e), (y, G, \omega))$

Figure 3.3: Véron key generation algorithm.

the secret key.

The key generation algorithm and the ID protocol parts of Véron's scheme are described, respectively, in Figure 3.3 and Figure 3.4.

3.3 CVE Scheme

To our knowledge, amongst all ID schemes whose security does not depend upon some number theoretic assumptions, only three of them involve five-pass, have a soundness error bounded by 1/2, and deal with values over a finite field $\mathbb{F}_q$ ($q > 2$): Chen's scheme [24], and the schemes based on Permuted Kernels (PKP) [76] and Constrained Linear Equations (CLE) [82]. Stern's five-pass variant is on a binary field, Permuted Perceptrons (PPP) [67] five-pass variant has a soundness

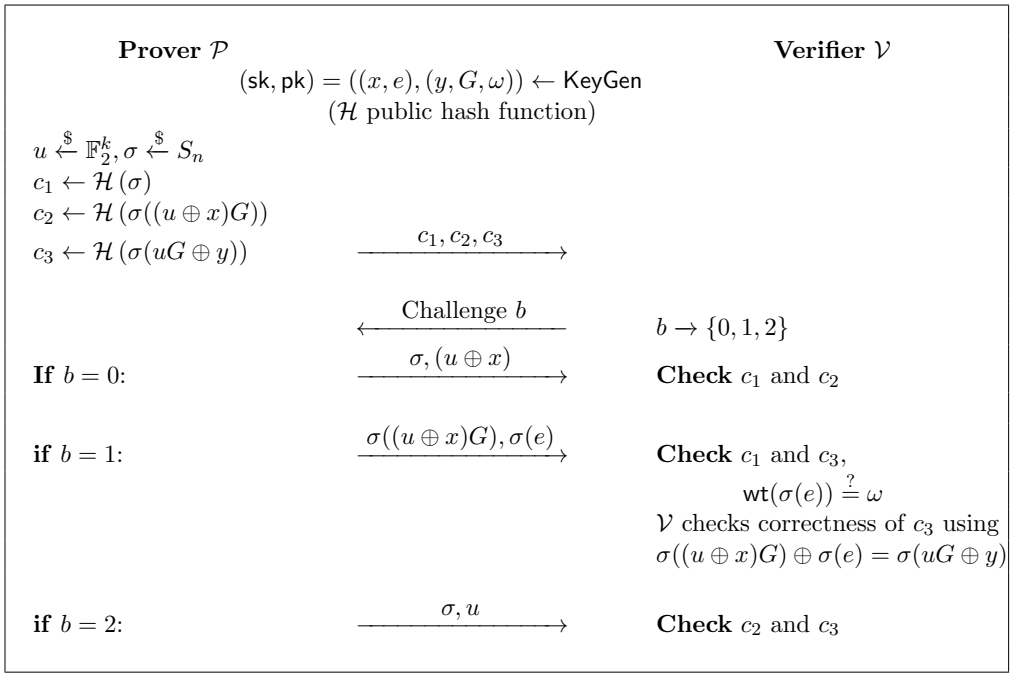

Figure 3.4: Véron identification protocol.

error bounded by 2/3 and $\mathcal{MQ}^*$-IP is a two-pass protocol [90].

Chen's scheme and those based on PKP and CLE have one thing in common: once the commitments sent, the verifier sends a random challenge which is an element $\alpha \in \mathbb{F}_q$. Then the prover sends back his secret vector scrambled by: a random vector, a random permutation and the value α.

Our proposal shows how to adapt this common step in the context of the syndrome decoding problem over $\mathbb{F}_q$ (qSD). Notice that while it is known since Barg's paper in 1994, that the qSD problem is NP-hard, it's only from the works developed in [55, 63], where the ISD algorithm is studied over $\mathbb{F}_q$, that it was possible to set up realistic parameters for the security of an ID scheme based on the qSD problem.

In what follows, we write elements of $\mathbb{F}_q^n$ as n blocks of size $\lceil \log_2(q) \rceil = N$. We represent each element of $\mathbb{F}_q$ as N bits. We first introduce a special transformation that we use in our protocol.

Definition 3.1. Let Σ be a permutation of $\{1, \ldots, n\}$ and $\gamma = (\gamma_1, \ldots, \gamma_n) \in \mathbb{F}_q^n$ such that $\forall i, \gamma_i \neq 0$. We define the transformation $\Pi_{\gamma,\Sigma}$ as :

$$\begin{array}{rccc} \Pi_{\gamma,\Sigma}: & \mathbb{F}_q^n & \longrightarrow & \mathbb{F}_q^n \\ & v & \mapsto & (\gamma_{\Sigma(1)} v_{\Sigma(1)}, \ldots, \gamma_{\Sigma(n)} v_{\Sigma(n)}) \end{array}$$

Notice that $\forall \alpha \in \mathbb{F}_q$, $\forall v \in \mathbb{F}_q^n$, $\Pi_{\gamma,\Sigma}(\alpha v) = \alpha \Pi_{\gamma,\Sigma}(v)$, and $\mathsf{wt}(\Pi_{\gamma,\Sigma}(v)) = \mathsf{wt}(v)$.

Our proposal (the CVE scheme) consists of two parts: a key generation algorithm and an ID protocol described respectively in Figure 3.5 and Figure 3.6. In the following we describe these two parts.

KeyGen:
Choose n, k, ω, and q such that $\mathrm{WF}_{\mathrm{ISD}}(n, r, \omega, q) \geq 2^{\kappa}$
$H \xleftarrow{\$} \mathbb{F}_q^{r \times n}$
$s \xleftarrow{\$} \mathbb{F}_q^n$, s.t. $\mathsf{wt}(s) = \omega$.
$y \leftarrow Hs^T$
Output $(\mathsf{sk}, \mathsf{pk}) = (s, (y, H, \omega))$

Figure 3.5: CVE key generation algorithm.

CVE key generation algorithm Let κ be the security parameter and $n, r = n-k, \omega$ the secure chosen parameters. The CVE scheme uses a random $(r \times n)$ q-ary matrix H common to all users which can be considered to be the parity check matrix of a random linear $[n, k, \omega]$ q-ary code. We can assume that H is described as $(I_r|R)$ where R is a random $r \times r$ matrix; as Gaussian elimination does not change the code generated by H, there is no loss of generality. Figure 3.5 presents the key generation algorithm.

CVE protocol The secret key holder can prove his knowledge of s by using two blending factors: a random vector and a special transformation which has the advantage to hide the non-zero values of the secret s. In the next section we show how a dishonest prover not knowing s can cheat the verifier in the protocol with probability of $q/2(q-1)$; this is reasonably close to $1/2$ for big enough q. Thus, the protocol has to be run several times to detect cheating provers. The security of the CVE scheme relies on the hardness of the syndrome decoding problem defined over $\mathbb{F}_q$ (qSD), that is on the difficulty of determining the preimage s of $y = Hs^T$.

3.3.1 Properties and security analysis

Zero-knowledge-proof Let $I = (H, y, \omega)$ be the public data shared by the prover and the verifier in our construction, and let $P(I, s)$ be the predicate.

$P(I, s)$ = "s is a vector which satisfies $Hs^T = y, \mathsf{wt}(s) = \omega$". We show in this section that the protocol presented in Figure 3.6 corresponds to a zero-knowledge interactive proof. To this end, we provide in the following proofs for the completeness, soundness, and zero-knowledge properties of the CVE scheme.

Completeness Clearly, each honest prover which has the knowledge of a valid secret s, the blending mask u, and the permutation $\Pi_{\gamma,\Sigma}$ for the public data can answer correctly any of the honest verifier's queries in any given round, thus the completeness property of the scheme is satisfied.

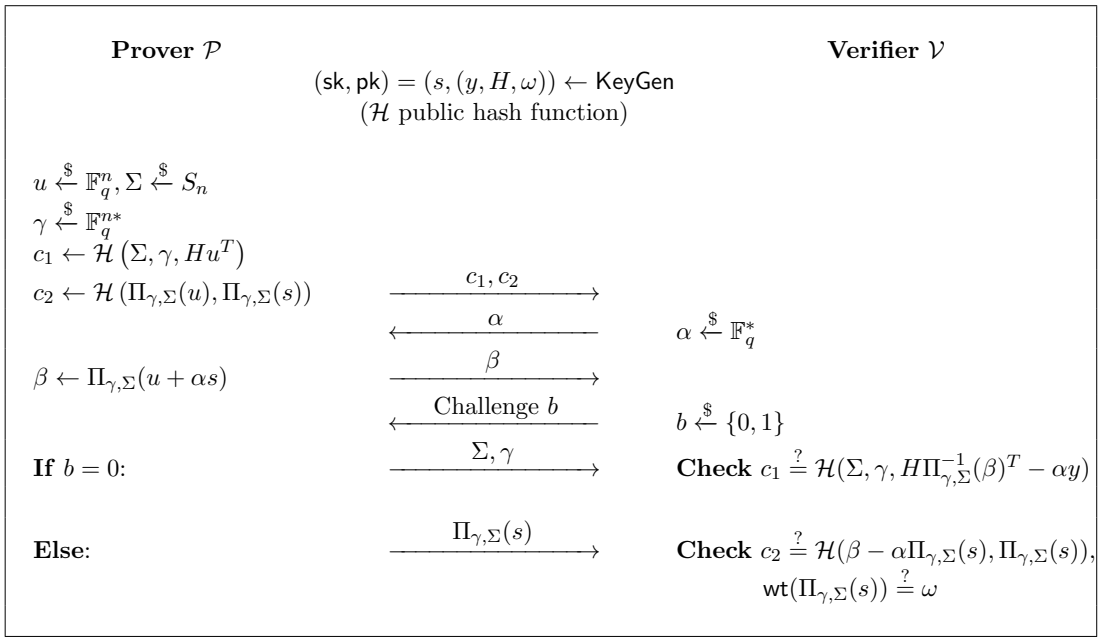

Figure 3.6: CVE identification protocol

Zero-Knowledge The zero-knowledge property for the CVE ID protocol is proved in the random-oracle model assuming that the hash function $\mathcal{H}$ has statistical independence properties.

Theorem 3.2. *The CVE ID protocol is a zero-knowledge interactive proof for $P(I, s)$ in the random-oracle model.*

Proof. The proof uses the classical idea of resettable simulation [44]. Let Sim be a polynomial-time probabilistic simulator (Turing machine) using a dishonest verifier. Because of the two interaction with the prover, we have to assume that the dishonest verifier could contrive two strategies : $St_1(c_1, c_2)$ taking as input the prover's commitments and generating a value $\alpha \in \mathbb{F}_q^*$, $St_2(c_1, c_2, \beta)$ taking as input the prover's commitments, the answer β and generating as output a challenge in the set $\{0, 1\}$. Sim generate a communication tape representing the interaction between prover and verifier. The goal is to produce a communication tape whose distribution is indistinguishable from a real tape by an honest interaction. The simulator Sim is constructed as follows :

Step 1. Sim randomly picks a query b from $\{0, 1\}$.

- If $b = 0$, Sim randomly chooses: u, γ, and Σ, and solves the equation: $Hs'^T = y$ for some s' not necessarily satisfying the condition $\mathsf{wt}(s') = \omega$. The commitments are taken as $c_1 = \mathcal{H}(\Sigma, \gamma, Hu^T)$, and c_2 as a random string. By simulating the verifier, Sim applies $St_1(c_1, c_2)$ to get $\alpha \in \mathbb{F}_q^*$, and then computes $\beta = \Pi_{\gamma,\Sigma}(u + \alpha s')$, and has the information needed

to derive the simulated communication data between prover and verifier. Therefore the candidates to be written in the communication tape consist of elements $A = c_1|c_2$, β and $ans = \gamma|\Sigma$. Taking into account the uniform distribution of the random variables used in the computation of A, ans and β, it follows that the distribution of these elements is indistinguishable from those resulting from a fair interaction.

- If $b = 1$, Sim also chooses u, γ, and Σ at random. This time it picks s as random from the set $\mathbb{F}_q^n$ with weight ω. The commitment c_1 is given uniformly at random value and $c_2 = \mathcal{H}(\Pi_{\gamma,\Sigma}(u), \Pi_{\gamma,\Sigma}(s))$. Again, from $St_1(c_1, c_2)$, Sim computes $\beta = \Pi_{\gamma,\Sigma}(u+\alpha s)$, and has the information needed to derive the simulated communication data. The communication set features elements $A = c_1|c_2$, β and $ans = \Pi_{\gamma,\Sigma}(s)$. The uniformly random character of the choices made render these elements indistinguishable from those resulting from a fair interaction.

Step 2. Sim applies the verifier's strategy $St_2(c_1, c_2, \beta)$ obtaining b' as result.

Step 3. When $b = b'$, Sim writes on its communication tape the values of A, α, β, b and ans. If the values differ, however, nothing is written and Sim returns to step 1.

Therefore, in 2δ rounds on average, Sim produces a communication tape indistinguishable from another that corresponds to a fair identification process execution that takes δ rounds. This concludes the proof.

□

Soundness: We now show that at each round, a dishonest prover is able to cheat a verifier to accept his identity with a probability limited by $q/(2(q-1))$.

Let us suppose that a dishonest prover has devised the following strategies to cope with the challenges that the verifier is expected to send. The first strategy (st_0) corresponds to the actions the prover takes when hoping to receive 0 as challenge. He chooses u, γ, and Σ at random and solves the equation $Hs'^T = y$ without satisfying the condition $\mathsf{wt}(s') = \omega$. Then he computes c_1 according to these values and randomly generates c_2. Thus, he is able to answer the challenge $b = 0$, regardless of the value of α chosen by the verifier. The second strategy (st_1) is successful in case a value 1 is received as challenge. He chooses u, γ and Σ at random and picks an s' with Hamming weight ω. With this choice, the commitment c_2 can be correctly reconstructed, and the Hamming weight of the fake private key validated. The commitment c_1 is randomly generated.

Now, these two strategies can be improved. Indeed a dishonest prover can try to make a guess on the value α sent by the verifier. Let α_c be the guessed value, so that β would be $\Pi_{\gamma,\Sigma}(u + \alpha_c s')$.

In st_0, instead of randomly generating c_2, he computes $c_2 = \mathcal{H}(\beta - \alpha_c\tilde{s}, \tilde{s})$ where $\tilde{s}$ is a random word of Hamming weight ω which will be sent as answer (if $b = 1$) instead of $\Pi_{\gamma,\Sigma}(s')$. With such a strategy, the cheater can answer to $b = 0$ regardless the value of α chosen by the verifier and to $b = 1$ if $\alpha = \alpha_c$.
In st_1, instead of randomly generating c_1, he computes $c_1 = \mathcal{H}(\Sigma, \gamma, Hu^T + \alpha_c(Hs'^T - y))$. With such a strategy, the cheater can answer to $b = 1$ regardless the value of α chosen by the verifier and to $b = 0$ if $\alpha = \alpha_c$.

Therefore, when we consider the probability space represented by the random variables b and α, the success probability of a strategy st for one round is given by:

$$\Pr[\text{cheating}] = \sum_{i=0}^{1} P(st = st_i)P(b = i) + P(st = st_i)P(b = 1 - i)P(\alpha = \alpha_c) = \frac{q}{2(q-1)}.$$

Though it was calculated for the particular strategies above, this value also corresponds to the upper limit for generic cheating strategies as shown below. The security assumptions that we make are as follows: we require that the commitment scheme be computationally binding and that the qSD problem be hard. We now show that if a cheating prover manages to answer more than $(\frac{q}{2(q-1)})^\delta$ of the queries made by a verifier after δ rounds, either of the security assumptions above was broken, as stated in the theorem below.

Let us denote by $\overline{\mathcal{V}}$ an honest verifier and by $\tilde{\mathcal{P}}$ a cheating prover.

Theorem 3.3. *If $\overline{\mathcal{V}}$ accepts $\tilde{\mathcal{P}}$ proof with probability $\geqslant (\frac{q}{2(q-1)})^\delta + \varepsilon$, then there exists a polynomial time probabilistic machine which, with overwhelming probability, either computes a valid secret s or finds a collision for the hash function.*

Proof. Let T be the execution tree of $(\tilde{\mathcal{P}}, \overline{\mathcal{V}})$ corresponding to all possible questions of the verifier when the adversary has a random tape RA. $\overline{B}$ may ask $2(q-1)$ possible questions at each stage. Each question is a couple (α, b) where $\alpha \in \mathbb{F}_q^*$ and $b \in \{0, 1\}$. First we are going to show that, unless a hash-collision has been found, a secret key s can be computed from a vertex with $q + 1$ sons. Then we show that a polynomial time Sim can find such a vertex in T with overwhelming probability.
Let V be a vertex with $q + 1$ sons. This corresponds to a situation where 2 commitments c_1, c_2 have been made and where the cheater has been able to answer to $q+1$ queries. That is to say that there exists $\alpha \neq \alpha'$ such that the cheater answered correctly to the queries $(\alpha, 0)$, $(\alpha, 1)$, $(\alpha', 0)$ and $(\alpha', 1)$. Now let :

- (β, Σ, γ) the answer sent for the query $(\alpha, 0)$,
- (β, z) the answer sent for the query $(\alpha, 1)$,
- $(\beta', \Sigma', \gamma')$ the answer sent for the query $(\alpha', 0)$,

- (β', z') the answer sent for the query $(\alpha', 1)$,

the value z (resp. z') represents the expected value $\Pi_{\gamma,\Sigma}(s)$, (resp. $\Pi_{\gamma',\Sigma'}(s)$), hence $\mathrm{wt}(z) = \omega$. Notice also that the same value β (resp. β') is used for $(\alpha, 0)$ and $(\alpha, 1)$ (resp. $(\alpha', 0)$ and $(\alpha', 1)$) since it is sent before the bit challenge b. Then, because commitment c_1 (resp. c_2) is consistent with both queries $(\alpha, 0)$ and $(\alpha', 0)$ (resp. $(\alpha, 1)$ and $(\alpha', 1)$), we have:

$$\mathcal{H}(\Sigma, \gamma, H\Pi^{-1}_{\gamma,\Sigma}(\beta)^T - \alpha y) = c_1 = \mathcal{H}(\Sigma', \gamma', H\Pi^{-1}_{\gamma',\Sigma'}(\beta')^T - \alpha' y)\,,$$

and

$$\mathcal{H}(\beta - \alpha z, z) = c_2 = \mathcal{H}(\beta' - \alpha' z', z')\,.$$

The equations are satisfied by finding collisions on the hash function or having the following equalities:

$$\begin{aligned} \Sigma &= \Sigma' \\ \gamma &= \gamma' \\ z &= z' \\ H\Pi^{-1}_{\gamma,\Sigma}(\beta)^T - \alpha y &= H\Pi^{-1}_{\gamma',\Sigma'}(\beta')^T - \alpha' y \\ \beta - \alpha z &= \beta' - \alpha' z'\,. \end{aligned}$$

Hence:

$$\begin{aligned} H\Pi^{-1}_{\gamma,\Sigma}(\beta - \beta')^T(\alpha - \alpha')^{-1} &= y \\ (\beta - \beta')^T(\alpha - \alpha')^{-1} &= z\,. \end{aligned}$$

Then:

$$H\Pi^{-1}_{\gamma,\Sigma}(z) = y\,.$$

Therefore, the value $s = \Pi^{-1}_{\gamma,\Sigma}(z)$ with $\mathsf{wt}(\Pi^{-1}_{\gamma,\Sigma}(z)) = \mathsf{wt}(z) = \omega$, obtained from the equalities above, constitutes a secret key that can be used to impersonate the real prover.

Now, the assumption implies that the probability for T to have a vertex with $q+1$ sons is at least ε. Indeed, let us consider RA the random tape where $\tilde{\mathcal{P}}$ randomly picks its values, and let Q bet the set $\mathbb{F}_q^* \times \{0, 1\}$. These two sets are considered as probability spaces both of them with the uniform distribution.
A triple $(c, \alpha, b) \in (RA \times Q)^\delta$ represents the commitments, answers and queries exchanged between $\tilde{\mathcal{P}}$ and $\bar{\mathcal{V}}$ during an identification process (c represents commitments and answers). We say that (c, α, b) is "valid", if the execution of $(\tilde{\mathcal{P}}, \bar{\mathcal{V}})$ leads to the success state.
Let V be the subset of $(RA \times Q)^\delta$ composed of all the valid triples. The hypothesis of the lemma means that:

$$\frac{\mathrm{card}(V)}{\mathrm{card}((RA \times Q)^\delta)} \geq \left(\frac{q}{2(q-1)}\right)^\delta + \varepsilon.$$

Let Ω_δ be a subset of RA^δ such that:

- If $c \in \Omega_\delta$, then $q^\delta + 1 \leq \text{card}\{(\alpha, b),\ (c, \alpha, b) \text{ be valid } \} \leq (2(q-1))^\delta$,
- If $c \in RA^\delta \setminus \Omega_\delta$, then $0 \leq \text{card}\{(\alpha, b),\ (c, \alpha, b) \text{ be valid } \} \leq q^\delta$.

Then, $V = \{\text{valid } (c, \alpha, b), c \in \Omega_\delta\} \cup \{\text{valid } (c, \alpha, b), c \in RA^\delta \setminus \Omega_\delta\}$, therefore :

$$\text{card}(V) \leqslant \text{card}(\Omega_\delta)(2(q-1))^\delta + (\text{card}(RA^\delta) - \text{card}(\Omega_\delta))q^\delta.$$

Thus

$$\begin{aligned} \frac{\text{card}(V)}{\text{card}((RA \times Q)^\delta)} &\leqslant \frac{\text{card}(\Omega_\delta)}{\text{card}(RA^\delta)} + q^\delta \left((2(q-1))^{-\delta} - \frac{\text{card}(\Omega_\delta)}{\text{card}((RA \times Q)^\delta)} \right) \\ &\leq \frac{\text{card}(\Omega_\delta)}{\text{card}(RA^\delta)} + \left(\frac{q}{2(q-1)} \right)^\delta. \end{aligned}$$

It follows that:

$$\frac{\text{card}(\Omega_\delta)}{\text{card}(RA^\delta)} \geq \varepsilon.$$

This shows that the probability that an intruder might answer to (at least) $q^\delta + 1$ of the verifier's queries, by choosing random values, is greater than ε. Now, if more than $q^\delta + 1$ queries are bypassed by an intruder then $T(RA)$ has at least $q^\delta + 1$ leaves, i.e. $T(RA)$ has at least a vertex with $q + 1$ sons.
So, by resetting $\tilde{\mathcal{P}}$ $\frac{1}{\varepsilon}$ times, and by repeating again, it is possible to find an execution tree with a vertex with $q + 1$ sons with probability arbitrary close to one. This theorem implies that either the hash function $\mathcal{H}$ is not collision free , or the qSD problem is not intractable. Therefore, the soundness property was demonstrated, given that one must have the probability negligibly close to $1/2$. □

Proposed parameters

As for binary Stern's and Véron's ID schemes, the security of the CVE scheme relies on three properties of random linear q-ary codes:

1. Random linear codes satisfy the q-ary Gilbert-Varshamov lower bound [51];
2. For large n almost all linear codes lie over the Gilbert-Varshamov bound [64];
3. Solving the q-ary syndrome decoding problem for random codes is NP-hard [6].

We now have to choose parameters for an instantiation of the CVE scheme. We take into account the bounds corresponding to the Information Set Decoding algorithm over $\mathbb{F}_q$ and propose parameters for a security level of at least 80-bit security. The number of rounds must then be chosen in order to minimize the success probability of a cheater.

Let N be the number of bits needed to encode an element of $\mathbb{F}_q$, $\ell_{\mathcal{H}}$ the output size of the hash function $\mathcal{H}$, ℓ_Σ *(resp.* ℓ_γ*)* the size of the seed used to generate the permutation Σ *(resp. the vector* γ*)* via a pseudo-random generator, and δ the number of rounds. We have the following properties for the CVE scheme:

Size of the matrix in bits :

$$k \times k \times N (\text{we use the systematic form of } H)$$

Size of the public identification :

$$kN$$

Size of the secret key :

$$nN$$

Total number of bits exchanged:

$$\delta(2\ell_{\mathcal{H}} + N + nN + 1 + (\ell_\Sigma + \ell_\gamma + nN)/2)$$

Prover's computation complexity over $\mathbb{F}_q$:

$$\delta((k^2 + \mathsf{wt}(s)) \text{ multiplications} + (k^2 + \mathsf{wt}(s)) \text{ additions})$$

To obtain a precise complexity on the workfactor of ISD algorithms over $\mathbb{F}_q$ we've used the code developed by C. Peters, which estimates the number of iterations needed for an attack using a Markov chain implementation [63]. ISD algorithms depend on a set of parameters and this code allows to test which ones can minimize the complexity of the attack.

Since we use random linear codes, the syndrome decoding problem is hardest to solve when $k = n/2$ and ω is chosen slightly below the Gilbert-Varshamov bound (see Chapter 2). For the CVE scheme, we suggest the following parameters in order to reach 80 bit security according to the ISD algorithm:

$$q = 256, n = 128, k = 64, \mathsf{wt}(s) = 49.$$

For the same security level in Stern's and Véron's schemes, we need to take $n = 700, k = 350, \mathsf{wt}(s) = 75$.

To obtain 128 bit security, we have to choose these parameters,

$$q = 256, n = 208, k = 104, \mathsf{wt}(s) = 78,$$

which gives a scheme with the following properties:

Number of Rounds : 16
Matrix size (bits) : 86528
Public Id (bits) : 832
Secret key (bits) : 1664
Communication (bits) : 47248
Prover's Computation : $2^{17.4}$mult. and $2^{17.4}$add. over $\mathbb{F}_{256}$

	Stern	**Véron**	**Stern 5-pass**	**CVE**
Rounds	28	28	16	16
Matrix size (bits)	122500	122500	122500	32768
Public Id (bits)	350	700	2450	512
Secret key (bits)	700	1050	4900	1024
Communication (bits)	42019	35486	62272	31888
Prover's Computation (op.)	$2^{22.7}$	$2^{22.7}$	$2^{21.92}$	2^{16}
Over the Field	$\mathbb{F}_2$	$\mathbb{F}_2$	$\mathbb{F}_2$	$\mathbb{F}_{256}$

Table 3.1: CVE vs. Stern and Véron schemes.

Some improvements of the CVE scheme:

- To get better communication complexity in comparison to the version of the protocol presented in Figure 3.6, the prover could use a public function ϕ_q by sending $\phi_q^{-1}(\Pi_{\gamma,\Sigma}(s))$ instead of $\Pi_{\gamma,\Sigma}(s)$, where ϕ_q is an efficient bijective encoding which takes its input from the interval $[0, (q-1)^{\omega}\binom{n}{\omega}[$ and outputs a binary word of length n and Hamming weight ω. This function is described in Algorithm 6.1 (see Appendix).
- We could use the same random seed to generate the permutation Σ and the vector γ in order to further reduce the communication complexity.

Remark 3.4. To be fair, the two improvements of the CVE scheme above-mentioned are not taken into account in the calculation of the communication complexity in Table 3.1, since these improvements can be applied to Stern's and Véron's schemes as well. By using the same seed for Σ and γ, the communication complexity will be 30864 bits instead of 31888, and only 26760 bits if we use in addition an encoding function.

3.3.2 CVE vs. Stern's and Véron's schemes

In [81], Stern has proposed two five-pass variants of his scheme. The first one to lower the computing load. However, this variant slightly increases the cheating probability rather than lowering it, and thus increases the communication complexity. The other one minimizes the number of rounds and lowers the cheating probability to $(1/2)^{\delta}$.

Table 3.1 shows the advantage regarding the communication complexity and the size of the matrix of the CVE scheme in comparison to Stern's initial proposal, his second variant, and Véron's scheme, for 80 bit security and a cheating probability of 2^{-16}. We considered that all seeds used are 128 bits long and that hash values are 160 bits long.

	SH	**CLRS**	**AGS**	**CVE**
Rounds	16	16	18	16
Matrix size	45×30	64×2049	1×350	64×64
over the field	$\mathbb{F}_{2^4}$	$\mathbb{Z}$	$\mathbb{F}_2$	$\mathbb{F}_{256}$
Public Id (bits)	492	288	700	512
Secret key (bits)	2048	192	700	1024
Communication (bits)	13282	223104	20080	31888
Prover's computation (op.)	2^{22}	2^{16}	2^{21}	2^{16}
Area	Multivariates	Lattices	Codes	Codes

Table 3.2: CVE vs. other post-quantum schemes.

3.3.3 CVE vs. efficient post-quantum ID schemes

In this section we compare the CVE scheme to some other five-pass post-quantum identifications schemes. The first one was proposed by Sakumoto et al. at Crypto 2011, it a five-pass ID scheme based on multivariate quadratic polynomials [73]. The second one is similar to our construction, it was proposed by Cayrel et al. in [18] and its security is based on the hardness of the SIS lattice problem. The last one was proposed recently by Aguilar Melchor et al. [2]. They designed a code-based five-pass ID scheme (AGS) with an asymptotic soundness error of 1/2. However, at the difference of the CVE scheme, their scheme uses structural codes which seems weaker in comparison to random codes due to possible structural attacks. Table 3.2 shows a comparison between the CVE scheme and all these schemes for 80-bit security and a cheating probability of 2^{-16}.

3.3.4 CVE vs. ID schemes based on other problems

We compare our scheme to some other zero-knowledge ID schemes cited in Section 3.3 and whose security does not depend upon number theoretic assumptions, and where the whole cheating probability is bounded by $(1/2)^{\delta}$ (except for PPP). We use some results given in the corresponding papers of these schemes [66, 67, 69, 48], and try to adapt parameters such that the security level has to be as near as possible to 80 bits for a fair comparison. Notice that for CLE, the result given in our table does not fit with what is given in [67] and [69]. Indeed, as mentioned in [82], the zero-knowledge property of the scheme can only be stated if two quantities ($S\sigma$ and $T\tau$) are public in addition to the public identification. For PPP, we consider the three-pass version instead of the five one because, as stated by the authors in [67], it is more efficient from a computational point of view and furthermore easier to implement. We do not consider for the prover's complexity the cost of the computation of hash values but the number of these values needed for the computation is mentioned in Table 3.3.

	PKP	CLE	PPP	CVE
Rounds	16	16	39	16
Matrix size	24×24	24×24	161×177	64×64
over the field	$\mathbb{F}_{251}$	$\mathbb{F}_{257}$	$\mathbb{F}_2$	$\mathbb{F}_{256}$
Public Id (bits)	384	288	245	512
Secret key (bits)	128	192	177	1024
Communication (bits)	13456	16528	51441	31888
Prover's computation (op.)	$2^{13.28}$	$2^{13.28}$	$2^{21.1}$	2^{16}
Number of hash values	2	2	8	2
Bit security	85	84	> 74	87

Table 3.3: CVE vs. schemes based on another problems.

Reducing a public key size As we have already mentioned in Section 2.2, there have been some proposals to use quasi-cyclic or quasi-dyadic codes in order to reduce the public key-size. In context of ID schemes, Gaborit et al. proposed a variation of the Stern ID scheme by using double circulant codes [35]. Using this variant, Cayrel et al. described in [23] an implementation of the Stern's scheme in low-resource devices. In this sens, besides random matrices we also give in the next section implementation results for Stern's, Véron's and the CVE schemes using quasi-cyclic and quasi-dyadic matrices. Recently, There have been several structural attacks against such constructions, the first attack presented by Gauthier et al. in [39] and the second attack is due to Faugère et al. [30]; these attacks extract the private key of some parameters of these variants. We should mention that schemes using binary codes are unaffected by such attacks.

3.4 Implementation Results

In order to demonstrate the improvement of the CVE scheme in terms of efficiency compared to Stern's and Véron's schemes, we provide in this section the results for our **C** implementation of the three schemes.

In our implementation the public matrices G and H are given in systematic form, i.e. $G = [I_k|R]$ and $H = [I_{n-k}|R]$ respectively ($r = n - k$ and $k = n/2$), where only the redundant part R is used.

For the generation of random vectors and hash values used in these schemes we deployed Keccak,[1]. We have chosen Keccak, because it can be used as a hash function and as a stream cipher at the same time. But note that it can be replaced by any other suitable scheme providing the necessary functionality.

Finally, all the tests have been carried out on an Intel(R) Core(TM)2 Duo CPU E8400@3.00GHz machine running Ubuntu/Linux 2.6.32-21. The source has been compiled with gcc 4.4.3. It assumes a 64-bit architecture.

[1] http://keccak.noekeon.org

Stern's and Véron's schemes

For the implementation we use as parameters $n = 768$, $k = 384$, and $\omega = 76$ for Stern's (resp. Véron's) scheme in order to reach 80-bit security and at the same time to satisfy some implementation constraints. Double circulant and dyadic submatrices of H have a size of 64×64 bits. For instance, if $H \in \mathbb{F}_2^{384\times768}$ is quasi-cyclic or quasi-dyadic, then the submatrix R consists of $6 \times 6 = 36$ double circulant or dyadic submatrices of size 64×64 bits respectively. Due to the row-major order of $\mathbf{C}$, the product sH^T is more efficient as Hs^T ($s \in \mathbb{F}_2^n$). Hence, the implementation uses the transposed matrix H^T instead of H.

Memory requirements: the memory requirements for the Stern's and Véron's schemes are as follows: using a random matrix $384 \times 384 = 147.456$ bits are necessary to store the redundancy part R of H resp. G. Using quasi-cyclic (quasi-dyadic) matrices, the memory footprint for the matrices drops by a factor of 64. Only $6 \times 6 \times 64 = 2.304 = 147.456/64$ bits are needed. Hence, although the timings using quasi-cyclic (quasi-dyadic) matrices are worse than for random matrices, in some environments the smaller memory footprint might compensate for the loss in performance.

CVE scheme

It uses a parity check matrix H of size $r \times n$ over $\mathbb{F}_q$, where $q = 2^m$, $1 \leq m \leq 16$, $r = n - k$ and $k = n/2$. If H is quasi-cyclic or quasi-dyadic, then the submatrix R would consist of 81 double circulant or dyadic submatrices of 8×8 field elements.

The matrix size is always measured in numbers of field elements. Each field element occupies invariably 2 bytes of memory. Strictly speaking, this would be necessary only in the case $m = 16$. However, using only the necessary bits would complicate the code and slow down the computation. In environments in which memory is a very valuable resource, this fact had to be taken into account. The parameters used in this implementation are: $n = 144, r = 72, \omega = 55$, and $q = 2^m = 2^8 = 256$.

Table 3.4 shows a comparison of the experimental results for running times that a prover needs to be identified using the three ID schemes, for 80 bit security and a cheating probability of 2^{-16}.

	Stern	**Véron**	**CVE**
Rounds	28	28	16
Random	1.047 ms	0.896 ms	0.580 ms
Quasi-Cyclic	0.959 ms	0.962 ms	0.710 ms
Quasi-Dyadic	1.506 ms	1.494 ms	0.648 ms

Table 3.4: Timing results for CVE, Stern's, and Véron's ID schemes.

Memory requirements for our scheme: using a random matrix, $72 \times 72 \times 2 = 10.368$ bytes are necessary to store the redundancy part R of H resp. G. Using quasi-cyclic (quasi-dyadic) matrices, the memory footprint for the matrices drops by a factor of 8, because in this case only $9 \times 9 \times 8 \times 2 = 1.296 = 10.368/8$ bytes are needed. Again, as with the Stern's and Véron's scheme, memory savings using the structured matrix types might be more important than the loss in runtime.

4 Extended Security Arguments for Signature Schemes

Digital signatures are an essential security technology in the modern world, for instance they are the cornerstones of software security, e-business, e-government, and many more applications. One of the ways to build a signature scheme is firstly to construct an identification protocol and then convert it to a signature scheme using the well-known Fiat-Shamir paradigm. This idea has gained great popularity since its introduction because it yields efficient signature schemes. At Eurocrypt 1996, Pointcheval and Stern presented a new reduction technique to obtain security arguments of such transformation in the random-oracle model. The main tool of their security proof is the well-known forking lemma. However, this proof only works for canonical (three-pass) honest verifier zero-knowledge identification schemes.

Throughout the recent years, a number of five-pass identification schemes have been proposed, examples of such schemes can be found in [18, 80, 19, 73, 2, 76]. These schemes have the advantage that they provide better communication complexity compared to canonical identification schemes in their corresponding area, indeed, they fall outside the original framework if we want to transform them to secure signature schemes. To the best of our knowledge, there is no work that deals with this issue.

In this chapter we propose a generalization of the Fiat-Shamir paradigm for identification schemes with multi-pass. Furthermore, we provide by extending the forking lemma the security proof of the resulting schemes what we called n-generic signature schemes. These include signature schemes that are derived from certain $(2n+1)$-pass identification schemes for $n \geq 2$.

This chapter is based on a joint work with Özgür Dagdelen, Pascal Véron, David Galindo, and Pierre-Louis Cayrel [29]. It was presented at the fifth International Conference on Cryptology, Africacrypt 2012, at Al Akhawayn University in Ifrane, Morocco.

4.1 Introduction

The focus of this chapter is on methodologies to prove the security of digital signature schemes. Thus, instead of providing security reductions from scratch, the goal is to provide security arguments for a class of signature schemes, as previously done for example in [1, 58, 65, 68, 25]. In particular, we aim at extending a pioneering work by Pointcheval and Stern [65] where a reduction technique was introduced to obtain security arguments for the so-called generic signature schemes. These security

arguments allow for simple proofs and for efficient signature schemes. Moreover, this type of signature schemes can be derived from ID schemes through the Fiat-Shamir paradigm if the latter satisfy certain requirements.

The content of this chapter is organized as follows. We present a short description of the original Fiat-Shamir transformation with a security argument in Section 4.2. Afterwards, we show in Section 4.3 how to generalize the original Fiat-Shamir paradigm for identification scheme with multi-pass. In Section 4.4 we provide the extended security arguments for n-generic signature schemes, and we exemplify this by deriving a provably secure signature scheme based on the CVE scheme proposed in Chapter 3. In Section 4.5 we give implementation results of the signature schemes derived from the Stern's, Véron's, and CVE schemes using the Fiat-Shamir paradigm and its extended version.

4.2 Fiat-Shamir Paradigm and Security Argument

At Crypto 1986, Fiat and Shamir proposed a method to transform canonical ID schemes to signature schemes [32]. More precisely, this transformation works as follows. Let consider a canonical ID scheme where a prover sends first a commitment Com, then receives a challenge Ch drawn from a uniform distribution, and finishes the interaction with a message, called response Rsp. Finally, the verifier applies a verifying algorithm to the prover's public key, determining acceptance or rejection. In order to transform the ID scheme described above into a non-interactive signature scheme, Fiat-Shamir proposed the following approach. Since the (honest) verifier is supposed to draw the challenge Ch randomly and does not depend on the previous commitment Com by the prover, this step is replaced by introducing a random oracle which upon input the message M and commitment Com returns the challenge Ch to the prover. Now, the prover can compute the response Rsp given from Com and Ch. The obtained signature scheme of a message M, which called generic signature scheme is defined as follows $\sigma = (\sigma_0, h, \sigma_1) = (\mathsf{Com}, \mathsf{Ch}, \mathsf{Rsp})$. In order to use these signature schemes in practice, the authors of [32] suggested to instantiate the oracle by an appropriate hash function.

Pointcheval and Stern [65] provided security arguments for generic signature schemes. However, these generic signature schemes are restrictive in the sense that (a) they allow transformations only based on canonical ID schemes, and (b) they additionally enjoy the existence of a polynomial-time algorithm, called extractor, that recovers the signing key from two related signatures $\sigma = (\sigma_0, h, \sigma_1)$ and $\sigma' = (\sigma_0, h', \sigma_1')$ with $h \neq h'$. The main tool of the security proof proposed by Pointcheval and Stern is a forking lemma. This lemma states that a successful forger can be restarted with a different random oracle in order to get two distinct but related forgeries. Using this lemma the security of generic signature schemes is guaranteed under a supposedly intractable problem.

Remark. The work of Abdalla *et al.* [1] introduced a new transformation from ID schemes to signature schemes without insisting on the existence of such an extractor. Nonetheless, they require again canonical ID schemes. Ohta and Okamoto [58] assume that the ID scheme is honest-verifier (perfect) zero-knowledge and that it is computationally infeasible for a cheating prover to convince the verifier to accept. Again, this result is valid only for three-pass ID schemes.

Very recently, Yao and Zhao [25] presented what they call challenge-divided Fiat-Shamir paradigm. Here, security results are set for three-pass ID schemes with divided random challenges. Even though they consider more challenges, still ID schemes with more than three interactions are not captured by their paradigm.

4.2.1 Stern's signature scheme

We show in the following how canonical ID schemes can be turned to generic signature schemes through the original Fiat-Shamir paradigm, for instance we do this for the Stern's scheme presented in Chapter 3.

In order to obtain a signature $\mathrm{S} = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$ from the Stern's ID scheme we do the following.

Let δ be the number of rounds needed to achieve the required cheating probability.

$\mathsf{KGen}(1^\kappa)$ takes as input a security parameter 1^κ and outputs $\mathcal{K}(1^\kappa)$. The random oracle $\mathcal{O}$ outputs element of $\{0,1\}$, respectively.

$\mathsf{Sign}(\mathsf{sk}, M)$ takes as input sk (as defined in Figure 3.1) and a message M,

- computes $c_{0,i} = \mathcal{H}\left(\sigma_i | H u_i^T\right), c_{1,i} = \mathcal{H}\left(\sigma_i(u_i)\right)$, $c_{2,i} = \mathcal{H}\left(\sigma_i(u_i \oplus s)\right)$, and sets $\sigma_{0,i} = (c_{0,i}, c_{1,i}, c_{2,i})$, where $u_i \xleftarrow{\$} \mathbb{F}_2^n$ and $\sigma_i \xleftarrow{\$} S_n$, for all $1 \leq i \leq \delta$.
- computes $h = \mathcal{O}(M, \sigma_{0,1}, \ldots, \sigma_{0,\delta})$, with $h = (h_1, \ldots h_\delta) \in \{0,1\}^\delta$,
- sets $\sigma_{1,i} = (\sigma_i, u_i)$ if $h_i = 0$, $\sigma_{1,i} := (\sigma_i, u_i \oplus s)$ if $h_i = 1$, and $\sigma_{2,i} := (\sigma_i(u_i), \sigma_i(s))$ if $h_i = 2$, for all $1 \leq i \leq \delta$.

Finally, returns the signature σ for the message M as (σ_0, h, σ_1), where $\sigma_j = (\sigma_{j,1}, \ldots, \sigma_{j,\delta})$ with $0 \leq j \leq 1$.

$\mathsf{Vf}(\mathsf{pk}, M, \sigma)$ takes as input a public key pk (as defined in Figure 3.1), a message M and a signature σ, outputs 1 iff $(\sigma_{1,1}, \ldots, \sigma_{1,\delta})$ is well calculated as in the ID protocol.

Security Argument. Now, We prove that the signature scheme derived from the Stern's zero-knowledge ID scheme is secure against adaptively chosen-message attacks. We assume that an adversary produces a valid signature (σ_0, h, σ_1) for a message M. By applying the forking lemma introduced by Pointcheval and Stern we can find a second forgery $(\sigma_0, h', \sigma_1')$ with a non-negligible probability, such that $h \neq h'$. That leads to the existence of an index i with $1 \leq i \leq \delta$, such that $h_i \neq h_i'$. W.l.o.g. assume $h_i = 0$ and $h_i' = 1$. Now, the adversary gets the answers for two distinct challenges, namely (σ_i, u_i) and $(\sigma_i, u_i \oplus s)$. Finally, by XORing the two values

u_i and $(u_i \oplus s)$, the secret key s can be disclosed. This contradicts the intractability of the SD problem. The same result can be obtained for the two remaining cases, i.e. $h_i = 0$ and $h'_i = 2$ or $h_i = 1$ and $h'_i = 2$.

4.3 Generalized Fiat-Shamir Paradigm

Our goal in this section is to enlarge the class of ID protocols to which the Fiat-Shamir transformation can be applied. We identify a potential set of candidates that we name *n-canonical ID schemes*. By these schemes we mean schemes secure with respect to impersonation against passive attacks, where the challenges are drawn from an uniform distribution and have $(2n + 1)$-pass for $n \geq 2$. Such schemes can be defined as follows.

Definition 4.1 (n-canonical ID scheme)**.** An n-canonical ID scheme $\mathrm{ID} = (\mathcal{K}, \mathcal{P}, \mathcal{V})$ is a $(2n+1)$-pass interactive protocol. $\mathcal{K}$ and $\mathcal{P} = (\mathsf{P_1}, \ldots, \mathsf{P_{n+1}})$ are PPT algorithms whereas $\mathcal{V} = (\mathsf{ChSet}, \mathsf{Vf})$ with ChSet being a PPT algorithm and Vf a deterministic boolean algorithm. These algorithms are defined as follows:

$\mathcal{K}(1^\kappa)$ upon input a security parameter 1^κ, outputs a secret and public key $(\mathsf{sk}, \mathsf{pk})$ and challenge spaces $G_1, \ldots, G_n$ with $1/|G_i|$ negligible in 1^κ.

$\mathsf{P_1}(\mathsf{sk})$ upon input a secret key sk outputs the commitment R_1.

$\mathsf{P_i}(\mathsf{sk}, R_1, C_1, \ldots, R_{i-1}, C_{i-1})$ for $i = 2, \ldots, n$, upon input a secret key sk and the current transcript $R_1, C_1, \ldots, R_{i-1}, C_{i-1}$, outputs the i-th commitment R_i.

$\mathsf{P_{n+1}}(\mathsf{sk}, R_1, C_1, \ldots, R_n, C_n)$ upon input a secret key sk and the current transcript $R_1, C_1, \ldots, R_n, C_n$, outputs a response Rsp.

$\mathsf{ChSet}(\mathsf{pk}, i)$ upon input a public key pk and round number i, outputs a challenge $C_i \in G_i$.

$\mathsf{Vf}(\mathsf{pk}, R_1, C_1, \ldots, R_n, C_n, Rsp)$ upon input a public key pk, and the current transcript $R_1, C_1, \ldots, R_n, C_n, Rsp$, outputs either 1 (= valid) or 0 (= invalid).

An n-canonical ID scheme has the following properties.

Public-Coin For any index $i \in \{1, \ldots, n\}$ and any $(\mathsf{sk}, \mathsf{pk}, G_1, \ldots, G_n) \leftarrow \mathcal{K}(1^\kappa)$ the challenge $C_i \leftarrow \mathsf{ChSet}(\mathsf{pk}, i)$ is uniform in G_i.

Honest-Verifier Zero-Knowledge There exists a PPT algorithm Z, the zero-knowledge simulator, such that for any pair of PPT algorithms $D = (D_0, D_1)$ the following distributions are computationally indistinguishable:

- Let $(\mathsf{pk}, \mathsf{sk}, state) \leftarrow D_0(1^\kappa)$, and $trans = (R_1, C_1, \ldots, R_n, C_n, Rsp) \leftarrow \langle \mathcal{P}(\mathsf{sk}, \mathsf{pk}), \mathcal{V}(\mathsf{pk}) \rangle$ if pk belongs to sk, and otherwise $trans \leftarrow \perp$. Output $D_1(trans, state)$.

- Let $(\mathsf{pk}, \mathsf{sk}, state) \leftarrow D_0(1^\kappa)$, and $trans = (R_1, C_1, \ldots, R_n, C_n, Rsp) \leftarrow Z(\mathsf{pk}, 1)$ if pk belongs to sk, and otherwise $trans \leftarrow Z(\mathsf{pk}, 0)$. Output $D_1(trans, state)$.

Note that the definition of 1-canonical ID schemes is identical to that of canonical ID schemes [1].

4.3.1 From n-canonical ID to signature schemes

In this section we show how we can transform the n-canonical ID scheme defined above in order to get what we called n-generic signature scheme. Like the original Fiat-Shamir transform, the idea of this transformation consists of replacing the uniformly random challenges of the verifier as set by ChSet in the ID scheme by the outputs of some secure hash functions $\mathcal{H}_i : \{0,1\}^* \rightarrow G_i$ modeled as random oracles. More precisely, let $\mathrm{ID} = (\mathcal{K}, \mathcal{P}, \mathcal{V})$ be an n-canonical ID scheme. The joint execution of $\mathcal{P}(\mathsf{sk}, \mathsf{pk})$ and $\mathcal{V}(\mathsf{pk})$ then defines an interactive protocol between the prover $\mathcal{P}$ and the verifier $\mathcal{V}$. At the end of the protocol $\mathcal{V}$ outputs a decision bit $b \in \{0,1\}$.

Let $\mathcal{H}_i$ denote a hash function with output of cardinality 2^{κ_i} (derived from the security parameter κ).

n-Generic Signature Scheme. We call the resulting signature scheme $\mathrm{S} = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$ derived from the ID scheme $\mathrm{ID} = (\mathcal{K}, \mathcal{P}, \mathcal{V})$ n-generic signature Scheme and it can be constructed as follows.

$\mathsf{KGen}(1^\kappa)$ takes as input security parameter 1^κ and returns $\mathcal{K}(1^\kappa)$.

$\mathsf{Sign}(\mathsf{sk}, M)$ takes as input a secret key sk and a message M and returns the transcript $\langle \mathcal{P}(\mathsf{sk}, \mathsf{pk}), \mathcal{V}(\mathsf{pk}) \rangle$ as the signature σ, i.e.,

$$\sigma = (\sigma_0, h_1, \ldots, h_n, \sigma_n) = (R_1, C_1, \ldots, R_n, C_n, Rsp)$$

or simply $\sigma = (\sigma_0, \ldots, \sigma_n, h_1 \ldots, h_n) = (R_1, \ldots, R_n, Rsp, C_1, \ldots, C_n)$. Here, C_i is defined by the equation $C_i := \mathrm{H}_i(M, R_1, \ldots, R_i, C_1, \ldots, C_{i-1})$. We require that the min-entropy of the random variable which outputs $\sigma_0, \ldots, \sigma_{n-1}$ must be in $\omega(|\mathcal{H}_n|)$.[1]

$\mathsf{Vf}(\mathsf{pk}, M, \sigma)$ takes as input a public key pk, a message M and a signature σ and returns $\mathcal{V}.\mathsf{Vf}(\mathsf{pk}, M, \sigma)$[2] as the decision bit.

Similar to generic signature schemes defined by Pointcheval and Stern [65] we require in the security proof from n-generic signature schemes a property which we call n-soundness. Informally, n-soundness means that the secret key can be extracted from

[1] This requirement is necessary so that our security arguments in Lemma 4.4 goes through

[2] By $\mathcal{V}.\mathsf{Vf}(\mathsf{pk}, M, \sigma)$ we mean the verification algorithm performed by the verifier from the underlying ID scheme ID.

two correlated valid signatures $\sigma = (\sigma_0, h_1, \ldots, \sigma_{n-1}, h_n, \sigma_n)$ and $\sigma' = (\sigma_0, h_1, \ldots, \sigma_{n-1}, h'_n, \sigma'_n)$ with $h_n \neq h'_n$ in polynomial-time and with a non-negligible probability.

Definition 4.2 (n-Soundness)**.** Let $\mathrm{S} = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$ be an n-generic signature scheme. We call S n-sound if there exists a PPT algorithm K, the knowledge extractor, such that for any κ and M, any $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KGen}(1^\kappa)$, any $\sigma = (\sigma_0, h_1, \ldots, \sigma_{n-1}, h_n, \sigma_n)$ and $\sigma' = (\sigma_0, h_1, \ldots, \sigma_{n-1}, h'_n, \sigma'_n)$ with $\mathsf{Vf}(\mathsf{pk}, M, \sigma) = \mathsf{Vf}(\mathsf{pk}, M, \sigma') = 1$ and $h'_n \neq h_n$, we have $\mathsf{sk} \leftarrow K(\mathsf{pk}, \sigma, \sigma')$ with non-negligible probability.

The notion of special-soundness[3] and n-soundness coincide if $n = 1$.
We like to note that if one sets $\sigma_0^* = (\sigma_0, h_1, \ldots, \sigma_{n-1})$, then S satisfies n-soundness as long as the (identical) signature scheme outputting $(\sigma_0^*, h_n, \sigma_n)$ satisfies special-soundness. Nonetheless, we stick to n-soundness through this paper, to emphasize that the *last challenge* is needed to be different.

4.4 Security Arguments for n-generic Signature Schemes

In this section we extend the forking lemma in order to use it for proving that any n-generic signature scheme satisfying what we call n-soundness is existentially unforgeable in the random-oracle model.

4.4.1 Extended Forking Lemma

Pointcheval and Stern introduced in [65] the forking lemma as a technique to prove the security of some families of signature schemes, namely generic signature schemes with special-soundness. This well-known lemma is applied to get two forgeries for the same message using a replay attack, after that, the two forgeries could be used to solve some computational problem which is assumed to be intractable. We firstly provide the Extended Forking Lemma in the no-message attack model, then we show that a successful forger in the adaptive chosen-message attacks model implies a successful forger in the no-message attack model, as long as the honest-verifier zero-knowledge property holds. In this way, the n-generic signature scheme will be proved to be existentially unforgeable under chosen-message attacks, which is the standard level of security that a signature scheme should achieve.

For the forking lemma proofs, we need an important lemma called splitting lemma. Let X and Y be two sets, this lemma states that one can split a set X into two subsets, (a) a non-negligible subset Ω consisting of "good" x's which provides a non-negligible probability of success over y ($y \in Y$), and (b) its complement, consisting of "bad" x's.

[3] Actually, special-soundness is a notion belonging to ID schemes. However, since this property is quite similar to the required property of generic signature schemes, this concept is used for both cases in the literature.

Lemma 4.3 (Splitting Lemma)**.** Let A be a subset of $X \times Y$ such that $\Pr[A(x,y)] \geq \epsilon$, then there exist $\Omega \subset X$ such that

1. $\Pr[x \in \Omega] \geq \epsilon/2$ (ϵ is a positive integer)
2. whenever $a \in \Omega$, $\Pr[A(a,y)] \geq \epsilon/2$

See [68] for the proof.

No-Message Attack Model

Consider parameters $k_1, \ldots, k_n$ derived from security parameter κ.

Lemma 4.4. *Let* S *be an n-generic signature scheme with security parameter κ. Let $\mathcal{A}$ be a PPT Turing machine given only the public data as input. If $\mathcal{A}$ can find a valid signature $(\sigma_0, \ldots, \sigma_n, h_1, \ldots, h_n)$ for a message M with a non-negligible probability, after asking the n random oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$ polynomially often (in κ), then, a replay of this machine with the same random tape, the same first oracles $\mathcal{O}_1, \ldots, \mathcal{O}_{n-1}$ and a different last oracle $\mathcal{O}_n$, outputs two valid signatures $(\sigma_0, \ldots, \sigma_n, h_1, \ldots, h_n)$ and $(\sigma_0, \ldots, \sigma'_n, h_1, \ldots, h'_n)$ for the same message M with a non-negligible probability such that $h_n \neq h'_n$.*

Proof. We are given a no-message adversary $\mathcal{A}$, which is a PPT Turing machine with a random tape ω taken from a set R_ω. During the attack, $\mathcal{A}$ may ask $q_1, \ldots, q_n$ (polynomially bounded in κ) queries to random oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$ with $q_j^{(i)}$ denoting the j-query to oracle $\mathcal{O}_i$. We denote by $q_1^{(i)}, \ldots, q_{q_i}^{(i)}$ the q_i distinct queries to the random oracles $\mathcal{O}_i$ and let $r^{(i)} = (r_1^{(i)}, \ldots, r_{q_i}^{(i)})$ be the answers of $\mathcal{O}_i$, for $1 \leq i \leq n$. Let $S_i^{q_i}$ denote the set of all possible answers from $\mathcal{O}_i$, i.e., $\{r_1^{(i)}, \ldots, r_{q_i}^{(i)}\} \in S_i^{q_i}$. Furthermore, we denote by

$\mathcal{E}$ the event that $\mathcal{A}$ can produce a valid signature $(\sigma_0, \ldots, \sigma_n, h_1, \ldots, h_n)$ for message M by using random tape ω and the answers $r_1^{(i)}, \ldots, r_{q_i}^{(i)}$ for $i \leq n$. Note that a valid signature implies $h_i = \mathcal{O}_i(M, \sigma_0, h_1, \ldots, h_{i-1}, \sigma_{i-1})$.

$\mathcal{F}$ the event that $\mathcal{A}$ has queried the oracle $\mathcal{O}_n$ with input $(M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1})$, i.e.,

$$\exists j \leq q_n : \; q_j^{(n)} = (M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1}).$$

Accordingly, its complement $\neg\mathcal{F}$ denotes

$$\forall j \leq q_n : \; q_j^{(n)} \neq (M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1}).$$

By hypothesis of the lemma, the probability that event $\mathcal{E}$ occurs ($\Pr[\mathcal{E}]$), is non-negligible, i.e., there exists a polynomial function $T(\cdot)$ such that $\Pr[\mathcal{E}] \geq \frac{1}{T(\kappa)}$. We know that

$$\Pr[\mathcal{E}] = \Pr[\mathcal{E} \wedge \mathcal{F}] + \Pr[\mathcal{E} \wedge \neg\mathcal{F}]. \tag{4.1}$$

Furthermore, we get

$$\begin{aligned}
&\Pr\left[h_n = \mathcal{O}_n(M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1}) \wedge \neg\mathcal{F}\right] \\
&\quad = \Pr\left[h_n = \mathcal{O}_n(M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1}) \mid \neg\mathcal{F}\right] \cdot \Pr\left[\neg\mathcal{F}\right] \\
&\quad \leq \Pr\left[h_n = \mathcal{O}_n(M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1}) \mid \neg\mathcal{F}\right] \\
&\quad \leq \frac{1}{2^{k_n}},
\end{aligned}$$

because the output of $\mathcal{O}_n$ is unpredictable and $(M, \sigma_0, \ldots, \sigma_{n-1})$ has a high min-entropy given the definition of n-generic signature. The event $\mathcal{E}$ implies that $h_n = \mathcal{O}_n(M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1})$, and thus we get

$$\Pr\left[\mathcal{E} \wedge \neg\mathcal{F}\right] \leq \Pr\left[h_n = \mathcal{O}_n(M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1}) \wedge \neg\mathcal{F}\right] \leq \frac{1}{2^{k_n}} \tag{4.2}$$

Relations (4.1) and (4.2) lead to

$$\Pr\left[\mathcal{E} \wedge \ \mathcal{F}\right] \geq \frac{1}{T(\kappa)} - \frac{1}{2^{k_n}} \geq \frac{1}{T'(\kappa)} \tag{4.3}$$

Note that a polynomial $T'(\cdot)$ must exist since the difference between a non-negligible and negligible term is non-negligible. Therefore, $\exists l \leq q_n$ so that

$$\Pr\left[\mathcal{E} \wedge q_l^{(n)} = (M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1})\right] \geq \frac{1}{q_n T'(\kappa)}.$$

Indeed, if we suppose that, $\forall l \in \{1, \ldots, q_n\}$,

$$\Pr\left[\mathcal{E} \wedge q_l^{(n)} = (M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1})\right] < \frac{1}{q_n T'(\kappa)}$$

then,

$$\begin{aligned}
\Pr\left[\mathcal{E} \wedge \ \mathcal{F}\right] &= \Pr\left[\mathcal{E} \wedge (\exists j \leq q_n, q_j^{(n)} = (M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1}))\right] \\
&\leq \sum_{j=1}^{q_n} \Pr\left[\mathcal{E} \wedge q_j^{(n)} = (M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1})\right] \\
&< \frac{q_n}{q_n T'(\kappa)} = \frac{1}{T'(\kappa)}
\end{aligned}$$

This leads to a contradiction with (4.3). Further, we define

$$B = \{(\omega, r^{(1)}, \ldots, r^{(n)}) \text{ s.t. } \mathcal{E} \wedge q_l^{(n)} = (M, \sigma_0, h_1, \ldots, h_{n-1}, \sigma_{n-1})\}.$$

Since, $B \subset R_\omega \times S_1^{q_1} \times \ldots \times S_n^{q_n}$ and $\Pr[B] \geq \frac{1}{q_n T'(\kappa)}$, by using the splitting lemma we have:

- $\exists \Omega \subset R_\omega$ such that $\Pr[\omega \in \Omega] \geq \frac{1}{2q_n T'(\kappa)}$.
- $\forall \omega \in \Omega$, $\Pr\left[(\omega, r^{(1)}, \ldots, r^{(n)}) \in B\right] \geq \frac{1}{2q_n T'(\kappa)}$, where the probability is taken over $S_1^{q_1} \times \ldots \times S_n^{q_n}$.

We define

$$B' = \{(\omega, r^{(1)}, \ldots, r^{(n)}) \text{ s.t. } (\omega, r^{(1)}, \ldots, r^{(n)}) \in B \wedge \omega \in \Omega\}.$$

Recall that $r^{(i)} = (r_1^{(i)}, \ldots, r_{q_i}^{(i)})$ where $r_j^{(i)} \in S_i$ for $1 \leq j \leq q_i$. Since,

$$B' \subset (R_\omega \times S_1^{q_1} \times \ldots \times S_n^{l-1}) \times S_n^{q_n - l + 1},$$

by using the splitting lemma again we get

- $\exists \Omega' \subset R_\omega \times S_1^{q_1} \times \ldots \times S_n^{l-1}$ such that $\Pr\left[(\omega, r^{(1)}, \ldots, r^{(n-1)}, (r_1^{(n)}, \ldots, r_{l-1}^{(n)})) \in \Omega'\right] \geq \frac{1}{4q_n T'(\kappa)}$.
- $\forall (\omega, r^{(1)}, \ldots, r^{(n-1)}, (r_1^{(n)}, \ldots, r_{l-1}^{(n)})) \in \Omega'$, $\Pr\left[(\omega, r^{(1)}, \ldots, r^{(n-1)}, (r_1^{(n)}, \ldots, r_{l-1}^{(n)}, r_l^{(n)}, \ldots, r_{q_n}^{(n)})) \in B'\right] \geq \frac{1}{4q_n T'(\kappa)}$, where the probability is taken over $S_n^{q_n - l + 1}$.

As a result, if we choose l, ω, $(r^{(1)}, \ldots, r^{(n-1)}, (r_1^{(n)}, \ldots, r_{l-1}^{(n)}))$, $(r_l^{(n)}, \ldots, r_{q_n}^{(n)})$, and $(r'^{(n)}_l, \ldots, r'^{(n)}_{q_n})$ randomly, then we obtain two valid signatures $(\sigma_0, \ldots, \sigma_n, h_1, \ldots, h_n)$ and $(\sigma_0, \ldots, \sigma'_n, h_1, \ldots, h'_n)$ for message M with a non-negligible probability such that $h_n \neq h'_n$.[4] □

Chosen-Message Attack Model

We now provide the Extended Forking Lemma in the adaptively chosen-message attack model. In this model, an adversary may adaptively invoke a signing oracle and is successful if it manages to compute a signature on a new message. If the signing oracle outputs signatures which are indistinguishable from a genuine signer without knowing the signing key, then using the simulator one can obtain two distinct signatures with a suitable relation from a single signature, similarly to the no-message scenario.

Theorem 4.5 (The Chosen-Message Extended Forking Lemma)**.** *Let* S *be an n-generic signature scheme with security parameter κ. Let $\mathcal{A}$ be a PPT algorithm given only the public data as input. We assume that $\mathcal{A}$ can find a valid signature $(\sigma_0, \ldots, \sigma_n, h_1, \ldots, h_n)$ for message M with a non-negligible probability, after asking the n random oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$, and the signer polynomially often (in κ). Then,*

[4] Since l is the index of $\mathcal{A}$'s query and there are only polynomially number of queries made by $\mathcal{A}$, our success probability remains non-negligible when picking l randomly.

there exists another PPT algorithm $\mathcal{B}$ which has control over $\mathcal{A}$ by replacing interactions with the real signer by a simulation, and which provides with a non-negligible probability two valid signatures $(\sigma_0, \ldots, \sigma_n, h_1, \ldots, h_n)$ and $(\sigma_0, \ldots, \sigma'_n, h_1, \ldots, h'_n)$ for the same message M such that $h_n \neq h'_n$.

Proof. We consider a PPT algorithm $\mathcal{B}$ that executes $\mathcal{A}$ in such a way that $\mathcal{B}$ simulates the environment of $\mathcal{A}$. Therefore, $\mathcal{B}$ must simulate the interactions of $\mathcal{A}$ with random oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$ and with the real signer. Then, we could see $\mathcal{B}$ as an algorithm performing a no-message attack against the signature scheme S.

Let Sim denote the zero-knowledge simulator of S that can simulate the answers of the real signer without knowledge of the secret key and has access to the random oracles $\mathcal{O}_i$ $(1 \leq i \leq n)$. Let $\mathcal{A}$ be an adaptively chosen-message adversary, which is a probabilistic polynomial time Turing machine with a random tape ω taken from a set R_ω. During the attack, $\mathcal{A}$ may ask $q_1, \ldots, q_n$ queries to random oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$, and q_s queries (possibly repeated) to Sim. The values $q_1, \ldots, q_n$ and q_s are polynomially bounded in κ. We denote by $q_1^{(i)}, \ldots, q_{q_i}^{(i)}$ the q_i distinct queries to the random oracles $\mathcal{O}_i$, and by $M^{(1)}, \ldots, M^{(q_s)}$ the q_s queries to the simulator Sim.

The simulator Sim answers a tuple $(\sigma_0^{(j)}, \ldots, \sigma_n^{(j)}, h_1^{(j)}, \ldots, h_n^{(j)})$ as a signature for a message $M^{(j)}$, for each integer j with $1 \leq j \leq q_s$. Then, the adversary $\mathcal{A}$ assumes that $h_i^{(j)} = \mathcal{O}_i(M^{(j)}, \sigma_0^{(j)}, h_1^{(j)}, \ldots, h_{i-1}^{(j)}, \sigma_{i-1}^{(j)})$ holds for all $1 \leq i \leq n$ and $1 \leq j \leq q_s$, and stores all these relations.

Now we need to consider potential "collisions" of queries in the random oracles. There are two kind of collisions that can appear. That is, (a) the simulator Sim queries the random oracle with the same input the adversary has asked before (let us denote this event by $\mathcal{E}_1$), and (b) Sim asks the same question repeatedly (let us denote this event by $\mathcal{E}_2$).

We show that the probabilities of such events are negligible.

$$\begin{aligned}\Pr[\mathcal{E}_1] = \Pr[&\exists i \in \{1, \ldots, n\}; \exists j \in \{1, \ldots, q_s\}; \exists t \in \{1, \ldots, q_n\} | \\ &(M^{(j)}, \sigma_0^{(j)}, h_1^{(j)}, \ldots, h_{i-1}^{(j)}, \sigma_{i-1}^{(j)}) = q_t^{(i)}] \\ \leq \sum_{i=1}^{n} \sum_{j=1}^{q_s} \sum_{t=1}^{q_n} &\Pr[(M^{(j)}, \sigma_0^{(j)}, h_1^{(j)}, \ldots, h_{i-1}^{(j)}, \sigma_{i-1}^{(j)}) = q_t^{(i)}] \leq \frac{n q_s q_n}{2^\kappa},\end{aligned}$$

which is negligible, assuming that the σ_i's are random values drawn from a large set with cardinality greater than 2^κ.
Moreover, we have

$$\begin{aligned}\Pr[\mathcal{E}_2] = \Pr[&\exists i \in \{1, \ldots, n\}; \exists j, j' \in \{1, \ldots, q_s\} : j \neq j' | \\ &(M^{(j)}, \sigma_0^{(j)}, h_1^{(j)}, \ldots, h_{i-1}^{(j)}, \sigma_{i-1}^{(j)}) = (M^{(j')}, \sigma_0^{(j')}, h_1^{(j')}, \ldots, h_{i-1}^{(j')}, \sigma_{i-1}^{(j')})] \\ \leq \sum_{i=1}^{n} \sum_{j=1}^{q_s} \sum_{j'=1}^{j} &\Pr[(M^{(j)}, \sigma_0^{(j)}, h_1^{(j)}, \ldots, h_{i-1}^{(j)}, \sigma_{i-1}^{(j)}) = \\ &(M^{(j')}, \sigma_0^{(j')}, h_1^{(j')}, \ldots, h_{i-1}^{(j')}, \sigma_{i-1}^{(j')})] \leq \frac{n q_s^2}{2^\kappa},\end{aligned}$$

which is also negligible.

Algorithm $\mathcal{B}$ succeeds whenever the machine $\mathcal{A}$ produces a valid signature without any collisions. Hence, we have

$$\Pr[\mathcal{B} \text{ succeeds }] = \Pr[\mathcal{A} \text{ succeeds }] - \Pr[\mathcal{E}_1] - \Pr[\mathcal{E}_2] \geq \frac{1}{T(\kappa)} - \frac{nq_s q_n}{2^\kappa} - \frac{nq_s^2}{2^\kappa},$$

which is non-negligible.

Summing up, we have an algorithm $\mathcal{B}$ that performs a no-message attack against the signature scheme S in polynomial time with non-negligible probability of success. So we can use Lemma 4.4 applied to algorithm $\mathcal{B}$, and we obtain two valid signatures for the same message, such that $h_n \neq h'_n$ again in polynomial time.

□

Security of n-generic signature schemes

The following theorem states that all n-generic signature schemes satisfying n-soundness are existentially unforgeable under adaptively chosen-message attacks in the random-oracle model.

Theorem 4.6 (Security of n-Generic Signature Schemes)**.** *Let* S *be an n-generic signature scheme satisfying n-soundness with underlying hard problem* $\mathbf{P}$*. Let κ be the security parameter. Then,* S *is existentially unforgeable under adaptively chosen-message attacks.*

Proof. We assume that the underlying hardness $\mathbf{P}$ of the n-generic signature scheme is hard, i.e., for all PPT algorithms $\mathcal{A}$ the probability to solve a hard instance of $\mathbf{P}$ is negligible. The key generation algorithm KGen of S outputs a secret and public key pair $(\mathsf{sk}, \mathsf{pk})$ derived by a hard instance and its corresponding solution of the problem $\mathbf{P}$.

Now, assume by contradiction, that S is *not* existentially unforgeable under chosen-message attacks. That is, there exists a PPT algorithm $\mathcal{B}_1$ such that $\mathcal{B}_1$ is able to output a signature $\sigma^* = (\sigma_0, h_1, \ldots, \sigma_{n-1}, h_n, \sigma_n)$ for a fresh message M^* with non-negligible probability. Then, due to the Extended Forking Lemma, one can construct a PPT algorithm $\mathcal{B}_2$ which outputs two correlated signatures $\sigma^* = (\sigma_0, h_1, \ldots, \sigma_{n-1}, h_n, \sigma_n)$ and $\sigma^{**} = (\sigma_0, h_1, \ldots, \sigma_{n-1}, h'_n, \sigma'_n)$ with non-negligible probability such that $h_n \neq h'_n$.

Due to the n-soundness of S, we know that there exists an "extractor" which extracts the secret key given the two signatures above. This contradicts with the assumption that the underlying problem $\mathbf{P}$ is hard, and by implication, we learn that there cannot exist such a successful forger $\mathcal{B}_1$.

□

4.4.2 CVE signature scheme

Using our extended framework, we can show that all aforementioned five-pass ID schemes give raise to 2-generic signature schemes. We isolate a property, called n-

soundness, that implies unforgeability of all the schemes satisfying it. In this section we apply this for instance to the CVE scheme presented in Chapter 3. The resulting n-generic signature scheme $\mathrm{S} = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$ can be described as follows. Let δ be the number of rounds needed to achieve the required cheating probability.

$\mathsf{KGen}(1^\kappa)$ takes as input a security parameter 1^κ and outputs $\mathcal{K}(1^\kappa)$. The random oracles $\mathcal{O}_1$ and $\mathcal{O}_2$ output elements of $\mathbb{F}_q^\delta$ and $\{0,1\}^\delta$, respectively.

$\mathsf{Sign}(\mathsf{sk}, M)$ takes as input sk (as defined in Figure 3.5) and a message M,

- computes $c_{0,i} = \mathcal{H}\left(\Sigma_i, \gamma_i, Hu_i^T\right), c_{1,i} = \mathcal{H}\left(\Pi_{\gamma_i,\Sigma_i}(u_i), \Pi_{\gamma_i,\Sigma_i}(s)\right)$, sets $\sigma_{0,i} = (c_{0,i}, c_{1,i})$, where $u_i \xleftarrow{\$} \mathbb{F}_q^n, \Sigma_i \xleftarrow{\$} S_n$, and $\gamma_i \xleftarrow{\$} \mathbb{F}_q^{n*}$, for all $1 \le i \le \delta$,
- computes $h_1 = \mathcal{O}_1(M, \sigma_{0,1}, \ldots, \sigma_{0,\delta})$ with $h_1 = (h_{1,1}, \ldots h_{1,\delta}) \in \mathbb{F}_q^\delta$,
- computes $\sigma_{1,i} = \Pi_{\gamma_i,\Sigma_i}(u_i + h_{1,i}s)$,
- sets $h_2 = \mathcal{O}_2(M, \sigma_1, h_1, \sigma_2)$, where $\sigma_j = (\sigma_{j,1}, \ldots, \sigma_{j,\delta})$ with $0 \le j \le 1$, and $h_2 = (h_{2,1}, \ldots, h_{2,\delta}) \in \{0,1\}^\delta$,
- and finally, returns the signature σ for the message M as $(\sigma_0, h_1, \sigma_1, h_2, \sigma_2)$, where $\sigma_2 = (\sigma_{2,1}, \ldots, \sigma_{2,\delta})$ such that $\sigma_{2,i} = (\gamma_i, \Sigma_i)$ if $h_{2,i} = 0$ and, otherwise, $\sigma_{2,i} := \Pi_{\gamma_i,\Sigma_i}(s)$,

$\mathsf{Vf}(\mathsf{pk}, M, \sigma)$ takes as input a public key pk (as defined in Figure 3.5), a message M and a signature σ, and outputs 1 iff $(\sigma_{0,1}, \ldots, \sigma_{0,\delta})$ is well calculated as in the ID protocol, i.e., the following respective equation is valid for all $1 \le i \le \delta$:

$$\begin{aligned} \text{If } h_{2,i} = 0: \quad & c_{0,i} = \mathcal{H}(M, \Sigma_i, \gamma_i, H\Pi^{-1}_{\gamma_i,\Sigma_i}(\sigma_{1,i})^T - h_{1,i}y) \\ \text{If } h_{2,i} = 1: \quad & c_{1,i} = \mathcal{H}(M, \sigma_{1,i})^T - h_{1,i}\Pi_{\gamma_i,\Sigma_i}(s), \Pi_{\gamma_i,\Sigma_i}(s)) \\ & \wedge \quad \mathsf{wt}(\Pi_{\gamma,\Sigma}(s)) \stackrel{?}{=} \omega \end{aligned}$$

Security Argument. Using the Extended Forking Lemma, we prove in the following that the signature scheme derived from our zero-knowledge ID scheme is secure against adaptively chosen-message attacks. We assume that an adversary produces a valid signature $(\sigma_0, h_1, \sigma_1, h_2, \sigma_2)$ for a message M. By applying Theorem 4.5 we can find a second forgery $(\sigma_0, h_1, \sigma_1, h_2', \sigma_2')$ with a non-negligible probability, such that $h_2 \neq h_2'$. That leads to the existence of an index i with $1 \le i \le \delta$, such that $h_{2,i} \neq h_{2,i}'$. W.l.o.g. assume $h_{2,i} = 0$ and $h_{2,i}' = 1$. Now, the adversary gets the answers for two distinct challenges, namely (γ_i, Σ_i) and $\Pi_{\gamma_i,\Sigma_i}(s)$. Since we can construct Π_{γ_i,Σ_i} from (γ_i, Σ_i), the secret key s can be disclosed. This contradicts the intractability of the qSD problem.

4.5 Implementation Results

In this section we present the result of the implementation in **C** of the signature schemes derived from the Stern's, Véron's and CVE schemes using the Fiat-Shamir paradigm and its extending version.

Matrix Type	Dimension$_{[r\times n]}$	Weight	Time$_{[ms]}$		Msg.$_{[MiB]}$	Sec.$_{[bits]}$
			s	v		
Random	384×768	76	6.473	5.745	1	80
Quasi-cyclic	384×768	76	6.443	5.697	1	80
Quasi-dyadic	384×768	76	6.783	6.014	1	80

Table 4.1: Timing results for Stern's signature scheme.

Matrix Type	Dimension$_{[r\times n]}$	Weight	Time$_{[ms]}$		Msg.$_{[MiB]}$	Sec.$_{[bits]}$
			s	v		
Random	384×768	76	6.193	5.909	1	80
Quasi-cyclic	384×768	76	6.198	5.883	1	80
Quasi-dyadic	384×768	76	6.690	6.213	1	80

Table 4.2: Timing results for Véron's signature scheme.

All the tests have been carried out only on an Intel(R) Core(TM)2 Duo CPU E8400@3.00GHz machine running Linux 2.6.32-21 (Ubuntu). The implementation has been compiled with gcc 4.4.3, it assumes a 64-bit architecture.

The following tables show the runtime measurement of the three signature schemes; Stern's, Véron's and CVE signature schemes. For the signing and verification time (s/v) are used 140 rounds for Stern's and Véron's schemes, and 80 rounds for the CVE scheme in order to achieve 2^{-80} as cheating probability. We use the same parameters n, k, and ω as in Section 3.4, therefore the memory requirements for the three signature schemes is the same as for the corresponding ID schemes.

The signature size for Stern's and Véron's is about 25 KB and for the CVE scheme is about 19 KB for 80-bit security.

Matrix Type	Dimension$_{[r\times n]}$	Weight	Time$_{[ms]}$		Msg.$_{[MiB]}$	Sec.$_{[bits]}$
			s	v		
Random	72×144	55	2.683	2.354	1	80
Quasi-cyclic	72×144	55	2.947	2.133	1	80
Quasi-dyadic	72×144	55	2.869	2.142	1	80

Table 4.3: Timing results for the CVE signature scheme.

5 Threshold Ring Signature Schemes

Designing efficient threshold ring signature schemes is one of the recent hot research topics. Such schemes enable any t participating users belonging to a set of N users to produce a valid signature in such a way that the verifier cannot determine the identity of the actual signers. Some applications contexts, such as multi-user electronic shares, multi-user united elections and employee opinion survey, require the sharing of signing power in consideration of protecting the identities of signers.

The first code-based threshold ring signature scheme was proposed by Aguilar et al. in [3, 4]. Their proposal is a generalization of Stern's identification scheme. The major advantage of this construction is that its complexity depends linearly on a maximum number of signers N, comparing with the complexity of threshold ring signature schemes based on number theory whose complexity is $\mathcal{O}(tN)$. However, the disadvantage of large public key size and signature length is still unsolved for this scheme.

We present in this chapter a novel code-based threshold ring signature scheme based on the CVE scheme presented in chapter 3. Since this latter has a low soundness error allowing a specified security to be reached in few rounds, our construction uses this fact to achieve a secure scheme with shorter signature length, smaller public key size and signature cost compared to Aguilar et al.'s scheme. We confirm our results by providing implementation results in **C** for both schemes, which shows clearly the advantage of our proposal.

This chapter is based on joint work with Pierre-Louis Cayrel, Gerhard Hoffman, and Pascal Véron [17]. It was presented at the fourth International Workshop of Arithmetic of Finite Fields, WAIFI 2012, Bochum, Germany.

5.1 Introduction

The concept of ring signature schemes was introduced first in 2001 by Rivest et al. [72]. These schemes permit any user from a set of intended signers to sign a message with no existing group manager and to convince the verifier that the author of the signature belongs to this set without revealing any information about its identity.

In 2002, Bresson et al. [15] extended ring signature schemes in a (t, N)-threshold ring signature schemes, such schemes enable any t participating users belonging to a set of N users to produce a valid signature in such a way that the verifier cannot determine the identity of the actual signers. Some application scenarios are multi-user electronic shares, multi-user united elections and employee opinion survey. Bresson et al.'s scheme suffers from a lack of efficiency since the size of the signature grows with the number of users and the number of signers.

In [3, 4], Aguilar et al. introduced the first code-based threshold ring signature with complexity depending only on a maximum number of signers N. The main idea of this scheme is to generalize the Stern's ID scheme and convert this latter into a threshold ring signature scheme (TRSS) using the Fiat-Shamir paradigm. Aguilar et al.'s scheme was proven to be existentially unforgeable under a chosen-message attacks in the random oracle model and its security relies on the hardness of a variation of the syndrome decoding problem, called the binary Minimum Distance problem.

A second code-based TRSS was proposed by Dallot and Vergnaud in [28]. Their proposal is not derived from an ID scheme as opposed to Aguilar et al.'s scheme. Dallot and Vergnaud's scheme uses Goppa codes and combines the generic construction of Bresson et al. [15] and the CFS signature scheme [26]. This proposal has the advantage to provide a scheme having a short signature due to the use of CFS signature scheme. However, the public key size is too huge and the required time to generate a signature is too high. These disadvantages make it very difficult to use in practice.

The content of this chapter is organized as follows. First, we explain the basic idea how to construct a TRSS starting from an ID scheme in Section 5.2. Then, we present a short description of Aguilar et al's scheme in Section 5.3. Afterwards, we give in Section 5.4 a detailed description of our proposal, we discuss the security, and we show the advantage of our construction by giving a theoretical comparison with Aguilar et al.'s one and other similar post-quantum schemes. Finally, we show a performance aspect of our construction by providing implementation results.

5.2 TRSS from ID Schemes

In this section we show one of the way to construct a TRSS starting from an honest verifier zero knowledge ID scheme. For simplicity we consider only a canonical ID scheme but this idea holds for every ID scheme with arbitrary number of passes. Before to do this, we first introduce a formal definition of a (t, N)-threshold ring signature scheme together with the security requirements.

Definition 5.1. Let $t < N$ be integers. A (t, N)-threshold ring signature scheme consists of three algorithms:

$\mathsf{KGen}(1^\kappa)$ is a probabilistic algorithm which, on input a security parameter κ, outputs N pairs of private and public keys $(sk_1, pk_1), \ldots, (sk_N, pk_N)$.

$\mathsf{Sign}(\mathsf{sk}, M)$ is a probabilistic interactive protocol between t users, involving a set $(pk_1, \ldots, pk_N)$ of public keys, a set $(sk_{i1}, \ldots, sk_{it})$ of secret keys and a message M, and which outputs a (t, N)-threshold ring signature σ for the message M.

$\mathsf{Vf}(\mathsf{pk}, M, \sigma)$ is a deterministic algorithm which takes as input a threshold value t, a set of public keys $(pk_1, \ldots, pk_N)$ and a message/signature pair (M, σ), and

outputs 1 if σ is a valid (t, N)-threshold ring signature for the message M w.r.t. the public keys $(pk_1, \ldots, pk_N)$ and 0 otherwise.

For the security of a TRSS the basic criteria are:

- Unforgeability: Without the knowledge of the t secret keys, it is infeasible to generate a valid (t, N)-threshold ring signature.
- Anonymity: Given a message-signature pair, it should be infeasible for the verifier to reveal which t-subset of signers generated a signature.

See [4] for a formal definition of the two pervious properties.

A methodology for constructing a TRSS from an ID scheme consists of two main steps (a) and (b), which can be described as follows.

(a) From ID scheme to threshold ring identification scheme: in this step we extend a given ID scheme to a threshold ring identification scheme as follows.

We suppose that a set of t signers, one of them is the leader L, want to identify itself to a verifier. To do this, we perform the following algorithms:

(1) **Setup** takes a secret parameter as input and outputs the public parameters and chooses the leader.

(2) **Ring key generation** takes public parameters as input and outputs a pair of keys corresponding to the secret and the public keys.

(3) **Commitment-challenge-answer and verification step** is an interactive protocol between the t signers and the verifier consisting of the computation of the commitments, challenges and responses, following by a verification step which takes as input the answers of the challenges and verifies the honestly of the computation, and returns 1 (accept), and 0 (reject).
Figure 5.1 illustrates this step which can be described as follows.

- Each member of the t signers (including L) creates local commitments (com_i) using the secret keys and sends them to L.
- L collects the t values com_i, simulates the missing ones for the $(N - t)$ other users, and creates a commitment (COM), called a master commitment, which will be sent to the verifier.
- The verifier chooses randomly a challenge (Ch) and sends it to L who forwards it to the $(t - 1)$ signers.
- L collects the answers (Rsp_i) from the $(t - 1)$ signers, computes the responses for the $(N - t)$ users and finally computes a global answer (RSP) for the verifier.
- After receiving RSP, the verifier checks the correctness of the master commitment as in the underlying identification scheme.

The three algorithms (1), (2), and (3) constitute the obtained threshold ring identification scheme.

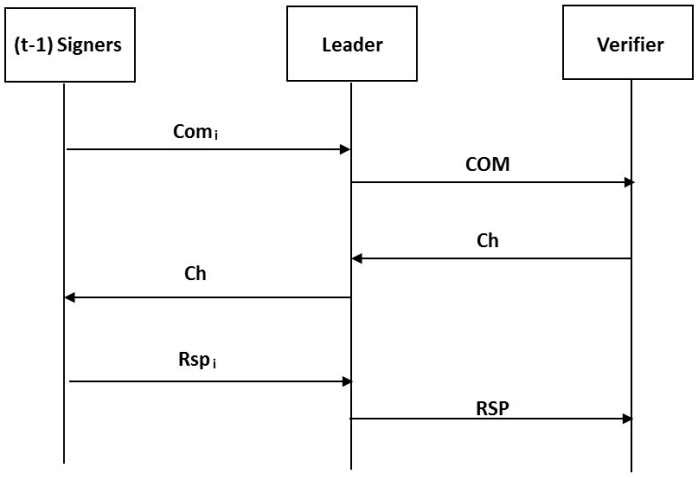

Figure 5.1: Threshold ring identification scheme.

(b) From threshold ring identification scheme to threshold ring signature scheme: by replacing a verifier with a secure hash function modeled as random oracle and then applying the Fiat-Shamir transform, as shown in chapter 4, we deduce a signature for a given message from a threshold ring identification scheme, this signature consists of the transcript of the interaction between the leader and the verifier.

5.3 Aguilar et al.'s TRSS

Aguilar et al. introduced in [3, 4] the first code-based TRSS, which is proved to satisfy the notion of unforgeablity and anonymity. The idea of the Aguilar et al's construction follows the methodology presented in the previous section. Starting from the Stern's ID scheme, we suppose that a set of t signers, one of them is the leader, interact in order to generate a signature of a given message. Each pair (signer$_i$, leader) executes the Stern's identification scheme, where signer$_i$ plays as prover and leader acts as verifier, sharing the same challenge. By this interaction, each signer uses a parity-check matrix H_i of a random code and a null syndrome as parameters. The use of syndromes equals to zero enables the leader to simulate the actions of the non-signers without knowing their secrets. On its turn, the pair (leader, verifier) runs an instance of Stern's identification scheme as well, where the commitments and answers are compositions involving the values received by the leader from the other signers. The leader applies block permutations over theses individual values in order to achieve the goal of anonymity. This process leads finally to obtain a Stern's threshold ring identification scheme, which was proven to be a zero-knowledge protocol with soundness error of 2/3 for each round as in the Stern's

ID protocol.

Finally, by using the Fiat-Shamir paradigm, the Stern's threshold ring identification scheme is transformed into a signature scheme. The signature size of this scheme is too huge, due to the fact that the Stern's threshold ring identification scheme has to be executed in multi-rounds in order to achieve a required cheating probability. Whereby, the number of rounds is constrained by the soundness error of the underlying identification protocol.

5.4 Our Proposal

The soundness error of approximately $1/2$ for the CVE scheme presented in chapter 3 allows a performance gains when compared to Stern's one. In order to make use of this gain, we present in this section an improved threshold ring identification scheme based on the CVE scheme.

To describe our scheme, we need the two notions of block permutations:

Definition 5.2. Let n and N be two integers and let $\beta = (\beta_1, \ldots, \beta_n, \beta_{n+1}, \ldots, \beta_{2n}, \ldots, \beta_{nN})$ be a vector of length nN defined over some alphabet. Let us define for $i \in [1, N]$ the elements $\tilde{\beta_i} = (\beta_{(i-1)n+1}, \ldots, \beta_{in})$ such that β can be expressed as $(\tilde{\beta_1}, \ldots, \tilde{\beta_N})$.

The constant (n, N)-block permutation Θ is a permutation over $\{1, \ldots, N\}$ which acts over vectors of length nN such that

$$\Theta(\beta) = \Theta(\tilde{\beta_1}, \ldots, \tilde{\beta_N}) = (\tilde{\beta}_{\Theta(1)}, \ldots, \tilde{\beta}_{\Theta(N)})$$

Let $\sigma = (\sigma_1, \ldots, \sigma_N)$ be a family of N permutations over $\{1, \ldots, n\}$, we define a (n, N)-block permutation Π, as a permutation which acts over a vector of length nN and which is the product of a constant n-block permutation Θ and the family σ, i.e.

$$\Pi(\beta) = \Theta(\sigma_1(\tilde{\beta_1}), \ldots, \sigma_N(\tilde{\beta_N}))$$

Roughly speaking, a constant (n, N)-block permutation divides a vector of length nN into N blocks of size n and permutes them. A (n, N)-block permutation permutes also for each block the components of the block.

Example 5.3. *The permutation* $(6, 5, 4, 3, 2, 1)$ *is* $(2, 3)$*-block permutation, and the permutation* $(3, 4, 5, 6, 1, 2)$ *is a constant* $(2, 3)$*-block permutation since the order on each block* $((1, 2), (3, 4)$ *and* $(5, 6))$ *is preserved in the block permutation.*

5.4.1 Description

We consider one set of N members. Let $(S_1, \ldots, S_t)$ be a subset of this set consisting of the members who want to prove that they know some secret s, whereas one of them is a leader L. The parameter t corresponding to the number of signers has to be fixed at the beginning of the protocol.

Our protocol consists of the following steps: Setup, Ring public key generation, Commitment-Challenge-Answer and Verification step. We describe each step as follows:

- **Setup** given κ as security parameter, we generate the corresponding public parameters n, r, ω, and q such that $\mathrm{WF}_{\mathrm{ISD}}(n, r, \omega, q) \geq 2^{\kappa}$, where n and r are the parameters for each matrix H_i $(1 \leq i \leq N)$ which will be used to form the ring public matrix. Each matrix can be constructed as follows: we choose a random vector $s_i \in \mathbb{F}_q^n$ of weight ω, generate $n - r - 1$ random vectors and consider the code $\mathcal{C}$ obtained by these $n - r$ words (the operation can be repeated until the co-dimension of $\mathcal{C}_i$ is r). The matrix H_i is then a parity-check matrix of a code $\mathcal{C}_i$ and thus we have $H_i s_i^T = 0$, where $s_i \in \mathbb{F}_q^n$ has a weight ω. The fact that we take a same syndrome and the same weight for the vectors s_i helps for conserving the anonymity in the group. For the $(N - t)$ other users, s_i are fixed at 0, because 0 is always a solution of the equation $H_i s_i^T = 0$.

- **Ring key generation** the leader collects all these matrices and forms among them the ring public matrix H, which can be described as follows:

$$H = \begin{pmatrix} H_1 & 0 & \cdots & 0 \\ 0 & H_2 & 0 & 0 \\ \vdots & \ddots & H_i & 0 \\ 0 & 0 & \cdots & H_N \end{pmatrix}$$

- **Commitment-challenge-answer and verification step** to simplify the description, we consider that the t signers correspond to the first matrices H_i $(1 \leq i \leq t)$. The leader L, member of the set of t signer among N members, want to prove to the verifier that he knows a secret key s, where s is a nN vector of weight tw. This will be achieved by performing the following steps:
  - Each member of the t signers (including L) creates local commitments using the secret keys s_i and sends them to L.
  - L collects all these commitments, simulates the missing ones for the $(N - t)$ other users by fixing all remaining s_i by 0, and creates the master commitment using a random constant block permutation.
  - The master commitment is sent to the verifier.
  - The verifier chooses a random value α over $\mathbb{F}_q$ and sends it to L, the latter one forwards this value to the $(t - 1)$ signers.
  - Each member of the t signers (including L) calculates the vectors β_i, L collects those values and creates a global vector β' using a constant block permutation and it will be sent to the verifier.
  - V chooses a challenge from $\{0, 1\}$ and sends it to L who forwards it to the $(t - 1)$ signers.

- L collects the answers from the $(t-1)$ signers, computes the responses for the other users and finally computes a global answer for the verifier.
- After receiving the global answer, the verifier checks the correctness of the master commitments.

Algorithm 5.1 gives a full description of this interaction between the set of t signers and the verifier. This algorithm has to be performed in multi-rounds in order to reach the required cheating probability.

We stress that during the answer step (line 19 of Algorithm 5.1), the knowledge of the permutation ρ permits to recover Θ, Σ_i, and γ_i for $1 \leq i \leq N$. In addition, the verifier can easily obtain β_i $(1 \leq i \leq N)$ by applying the inverse of θ on the known vector β'.

5.4.2 Security analysis

We prove that our threshold ring identification protocol is an honest-verifier zero-knowledge proof of knowledge.

Lemma 5.4. *Finding a vector s of length nN such that the global weight of s is tw, the weight of s for each of the N blocks of length n is 0 or ω, and such that s has a null syndrome for H, is hard under the assumption of hardness of the qMD problem.*

Proof. The construction of the matrix H (described above) and the vector s implies that finding such a n-block of length nN is also equivalent to finding a solution of a local hard problem s_i of weight ω such that $H_i s_i = 0$, which is hard under our assumption. □

Theorem 5.5. *Our scheme is an honest verifier zero-knowledge proof of knowledge, with soundness error bounded by $1/2$, that the group of t signers knows a vector s of length nN such that the global weight of s is $t\omega$, and such that the vector s has a null syndrome for H. The scheme is secure in the random-oracle model under the assumption of the hardness of the qMD problem.*

Proof. We prove that our scheme satisfies the three properties: completeness, soundness and zero-knowledge.

Completeness It is clear that each group of honest signers who has the knowledge of a valid secret key is able to answer correctly any of the honest leader's queries, which permit him to compute the master commitments. The leader, on his turn is able to reveal the information necessary to the honest verifier, in order to check the correctness of these commitments.

Algorithm 5.1 Generalized q-SD protocol

INPUT: $n, k, N, t \in \mathbb{N}$, where $k < n$ and $t < N$. $H \in \mathbb{F}_q^{rN \times nN}$, where $r = n - k$ and $\mathcal{H}$ a collision resistant hash function.

PRIVATE KEY: $s = (s_1, \ldots, s_N) \in \mathbb{F}_q^{nN}$, $\mathsf{wt}(s_j) = 0$ or ω, $\mathsf{wt}(s) = t\omega$, and $Hs^T = 0$.

COMMITMENT STEP:

Each signer S_i chooses $u_i \xleftarrow{\$} \mathbb{F}_q^n$, $\Sigma_i \xleftarrow{\$} S_n$, $\gamma_i \xleftarrow{\$} \mathbb{F}_q^{n*}$ $(1 \leq i \leq t)$.

S_i constructs $c_{1,i} \leftarrow \mathcal{H}\left(\Sigma_i | \gamma_i | H_i u_i^T\right)$ and $c_{2,i} \leftarrow \mathcal{H}\left(\Pi_{\gamma_i,\Sigma_i}(u_i) | \Pi_{\gamma_i,\Sigma_i}(s_i)\right)$.

S_i sends $c_{1,i}$ and $c_{2,i}$ to leader L.

L fixes the secret keys s_i of the $N - t$ other users at 0 $(t + 1 \leq i \leq N)$.

L chooses $N - t$ values $u_i \xleftarrow{\$} \mathbb{F}_q^n$ and $N - t$ permutations $\Sigma_i \xleftarrow{\$} S_n$ and $N - t$ values $\gamma_i \xleftarrow{\$} \mathbb{F}_q^{n*}$ $(t + 1 \leq i \leq N)$.

L chooses $\Theta \xleftarrow{\$} S_N$ in order to obtain the master commitments.

L computes the master commitments $C_1 \leftarrow \mathcal{H}(\Theta | c_{1,1} | \ldots | c_{1,N})$ and $C_2 \leftarrow \mathcal{H}(\Theta(c_{2,1}, \ldots, c_{2,N}))$.

C_1 and C_2 are sent to the verifier V.

V sends back the value $\alpha \xleftarrow{\$} \mathbb{F}_q$ and L passes it to each S_i $(1 \leq i \leq t)$.

S_i computes $\beta_i \leftarrow \Pi_{\gamma_i,\Sigma_i}(u_i + \alpha s_i)$ $(1 \leq i \leq t)$.

L computes $\beta_i \leftarrow \Pi_{\gamma_i,\Sigma_i}(u_i)$ $(t + 1 \leq i \leq N)$.

$\beta' = \Theta(\beta) = \Theta(\beta_1, \cdots, \beta_N) = (\beta_{\Theta(1)}, \ldots, \beta_{\Theta(N)})$ is sent to V.

CHALLENGE STEP:

V sends a challenge $b \xleftarrow{\$} \{0, 1\}$

ANSWER STEP: ▷ The first part of this step is between each signer S_i $(1 \leq i \leq t)$ and the leader L.

if $b = 0$ **then**

S_i sends γ_i and Σ_i to L.

else if $b = 1$ **then**

S_i sends $(\Pi_{\gamma_i,\Sigma_i}(s_i))$ to L.

end if

L simulates the $N - t$ other answers with $s_i = 0$ $(t + 1 \leq i \leq N)$.

L computes the answer for V:

if $b = 0$ **then**

$\gamma = (\gamma_1, \ldots, \gamma_N)$, $\Sigma = (\Sigma_1, \ldots \Sigma_N)$, and Θ are sent to V.

else if $b = 1$ **then**

$\rho(s) = (\Pi_{\gamma_{\Theta(1)},\Sigma_{\Theta(1)}}(s_{\Theta(1)}), \ldots, \Pi_{\gamma_{\Theta(N)},\Sigma_{\Theta(N)}}(s_{\Theta(N)}))$ is sent to V.

end if

VERIFICATION STEP:

if $b = 0$ **then**

V checks $C_1 \stackrel{?}{=} \mathcal{H}(\Theta | \mathcal{H}(\Sigma_1 | \gamma_1 | H_1 \Pi^{-1}_{\gamma_1,\Sigma_1}(\beta_1)^T) | \cdots | \mathcal{H}(\Sigma_N | \gamma_N | H_N \Pi^{-1}_{\gamma_N,\Sigma_N}(\beta_N)^T))$ and $\Theta \stackrel{?}{\in} S_N$.

else if $b = 1$ **then**

V checks

$$C_2 \stackrel{?}{=} \mathcal{H}\left(\begin{pmatrix} \mathcal{H}(\beta_{\Theta(1)} - \alpha\Pi_{\gamma_{\Theta(1)},\Sigma_{\Theta(1)}}(s_{\Theta(1)}) \mid \Pi_{\gamma_{\Theta(1)},\Sigma_{\Theta(1)}}(s_{\Theta(1)})) \\ \mathcal{H}(\beta_{\Theta(2)} - \alpha\Pi_{\gamma_{\Theta(2)},\Sigma_{\Theta(2)}}(s_{\Theta(2)}) \mid \Pi_{\gamma_{\Theta(2)},\Sigma_{\Theta(2)}}(s_{\Theta(2)})) \\ \vdots \\ \mathcal{H}(\beta_{\Theta(N)} - \alpha\Pi_{\gamma_{\Theta(N)},\Sigma_{\Theta(N)}}(s_{\Theta(N)}) \mid \Pi_{\gamma_{\Theta(N)},\Sigma_{\Theta(N)}}(s_{\Theta(N)})) \end{pmatrix}^T\right),$$

$\mathsf{wt}(\rho(s)) \stackrel{?}{=} t\omega$, and that $\rho(s)$ is formed of N blocks of length n and of weight ω or weight 0.

end if

Soundness It was proven in [19] that the underlying ID scheme satisfies this property and that the soundness error is bounded by $1/2$, assuming that the qMD problem is hard. Because our protocol can be seen as a composition of t simultaneous executions of the underlying ID scheme and given that the latter one can be reduced to our protocol by making all signing instances equal, this implies that this soundness error cannot be higher than $1/2$ for our protocol in one single round.

Zero-knowledge The zero-knowledge property for our protocol can be proven in the random-oracle model. In order to do that, we use the classical idea of resettable simulation. Let Sim be a polynomial-time probabilistic Turing machine (simulator) using a dishonest verifier. Because of the two interactions with the leader, we have to assume that the dishonest verifier could contrive two strategies: $St_1(C_1, C_2)$ taking as input the leader's (master) commitments and generating a value $\alpha \in \mathbb{F}_q$, $St_2(C_1, C_2, \beta')$ taking as input the leader's commitments, the answer β and generating as output a challenge in the set $\{0, 1\}$. Sim will generate a communication tape representing the interaction between leader and verifier. The goal is to produce a communication tape whose distribution is indistinguishable from a real tape by an honest interaction. The simulator Sim is constructed as follows:

Step 1. Sim randomly picks a query b from $\{0, 1\}$.

- If $b = 0$, Sim randomly chooses: u_i, γ_i, Σ_i $(1 \leq i \leq N)$ and Θ as a random constant block permutation on N blocks $\{1, 2, \ldots, N\}$, and solves the equation: $Hs'^T = 0$ for some vector $s' = (s'_1, \ldots, s'_N)$ of length nN and not necessarily satisfying the condition $\mathsf{wt}(s') = t\omega$. The values $c_{1,i}$ $(1 \leq i \leq N)$ can be computed as follows: $c_{1,i} = \mathcal{H}\left(\Sigma_i | \gamma_i | H_i u_i^T\right)$, the master commitments are taken then as $C_1 = \mathcal{H}(\Theta | c_{1,1} | \ldots | c_{1,N})$ and C_2 as a random string. By simulating the verifier, Sim applies $St_1(C_1, C_2)$ to get $\alpha \in \mathbb{F}_q$, and then computes β' as follows: $\beta' = \Theta(\Pi_{\gamma_1,\Sigma_1}(u_1 + \alpha s'_1), \ldots, \Pi_{\gamma_N,\Sigma_N}(u_N + \alpha s'_N))$, and has the information needed to derive the simulated communication data between leader and verifier. Therefore the candidates to be written in the communication tape consist of elements $A = C_1 | C_2$, β' and $ans = \rho = \Theta(\Pi_{\gamma_1,\Sigma_1}, \ldots, \Pi_{\gamma_N,\Sigma_N})$. Taking into account the uniform distribution of the random variables used in the computation of A, ans and β', it follows that the distribution of these elements is indistinguishable from those resulting from a fair interaction.
- If $b = 1$, Sim randomly chooses u_i, γ_i, Σ_i $(1 \leq i \leq N)$ and Θ as a random constant block permutation on N blocks $\{1, 2, \ldots, N\}$. This time it picks $s = (s_1, \ldots, s_N)$ as a random vector from the set $\mathbb{F}_q^{nN}$ with weight $t\omega$ and formed of N blocks of length n and of weight ω or 0. The commitments C_1 will be given uniformly at random values and $C_2 = \mathcal{H}(\Theta(c_{2,1}, \ldots, c_{2,N}))$ such that each $c_{2,i} = \mathcal{H}(\Pi_{\gamma_i,\Sigma_i}(u_i) | \Pi_{\gamma_i,\Sigma_i}(s_i))$. Again, from $St_1(C_1, C_2)$, Sim gets $\alpha \in \mathbb{F}_q$ and computes β' as follows: $\beta' = \Theta(\Pi_{\gamma_1,\Sigma_1}(u_1 + \alpha s_1), \ldots, \Pi_{\gamma_N,\Sigma_N}(u_N + \alpha s_N))$, and has the infor-

mation needed to derive the simulated communication data. The communication set features elements $A = C_1|C_2$, β' and $ans = \rho(s) = (\Pi_{\gamma_{\Theta(1)},\Sigma_{\Theta(1)}}(s_{\Theta(1)}), \ldots, \Pi_{\gamma_{\Theta(N)},\Sigma_{\Theta(N)}}(s_{\Theta(N)}))$. The uniformly random character of the choices made will render these elements indistinguishable from those resulting from a fair interaction.

Step 2. Sim applies the verifier's strategy obtaining b' as result.

Step 3. When $b = b'$, the machine Sim writes on its communication tape the values of A, α, β', b and ans. If the values differ, however, nothing is written and the machine returns to Step 1.

Therefore, in 2δ rounds on average, Sim produces a communication tape indistinguishable from one that corresponds to a fair interaction process execution that takes δ rounds. □

Now, using the generalized Fiat-Shamir paradigm presented in chapter 4, we can transform our honest verifier zero knowledge threshold ring identification protocol into a threshold ring signature scheme. The aim of the next section is to prove that the resulting threshold ring signature through this transformation is existentially unforgeable under chosen-message attacks in the random-oracle mode. We prove this result in general for ring signature derived from non-canonical ID scheme following the work of Herranz and Sáez [47] which require only canonical ID schemes as basis.

5.4.3 Extended security arguments for ring signature schemes

In this section we present a security proof for ring signatures obtained from non-canonical honest verifier zero-knowledge ID schemes. In order to do this, we generalize the work of [47]. This work provided a security proof for ring signature schemes obtained from canonical ID schemes. They achieve this by generalizing the forking lemma for a class of ring signatures which they call generic. This class is defined as follows. Consider a security parameter κ and a ring of r members $(S_1, \ldots S_r)$. Given a message M, its signature is formed by a tuple $(M, R_1, \ldots, R_r, h_1, \ldots, h_r, \eta)$, where $R_1, \ldots, R_r$ are randomly chosen values from a large set G, h_i is the output of a hash function $\mathcal{H}$ on input (M, R_i) for $1 \leq i \leq r$, and the value η is fully determined by the values $R_1, \ldots, R_r, h_1, \ldots, h_r$ and the message M.

Informally, the authors of [47] show given an adversary $\mathcal{A}$ which produces a signature $(M, R_1, \ldots, R_r, h_1, \ldots, h_r, \eta)$ within time T and success probability ε, there exists an adversary $\mathcal{B}$ which outputs two valid signatures $(M, R_1, \ldots, R_r, h_1, \ldots, h_r, \eta)$ and $(M, R_1, \ldots, R_r, h'_1, \ldots, h'_r, \eta')$ with $h_i \neq h'_i$ for some $1 \leq i \leq r$ with non-negligible probability ε' in time $T' \in \mathcal{O}(T\varepsilon'^{-1})$ by replaying $\mathcal{A}$ internally. Their result captures both no-message and adaptively chosen-message attacks.

In the following, we propose an extension of the forking lemma even more for a class of ring signatures schemes, which we call *n-generic*.

Let κ denote a security parameter (from which $G_1, \ldots, G_n$ are derived from) and n be an integer. Further, let $\mathcal{H}_i : \{0,1\}^* \to G_i$ denote hash functions for $1 \leq i \leq n$. We consider a ring $S_1, \ldots, S_r$ of r members. We capture n-generic ring signature in the following definition.

Definition 5.6 (n-Generic Ring Signature Scheme)**.** Assume the hash functions $\mathcal{H}_i$ are modeled by publicly accessible random oracles. An n-generic ring signature scheme is a ring signature scheme $\mathsf{RS} = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$ with the following properties:

Structure A signature σ for a message M is of the form $(\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_n, \mathbf{h}_1, \ldots, \mathbf{h}_n)$ where $\boldsymbol{\sigma}_0 = (\sigma_{0,1}, \ldots, \sigma_{0,r})$, $\boldsymbol{\sigma}_i = (\sigma_{i,1}, \ldots, \sigma_{i,r})$ and $\mathbf{h}_i = (h_{i,1}, \ldots, h_{i,r})$ for $i = 1, \ldots, n$. It holds that $h_{1,j} = \mathcal{H}_1(M, \sigma_{0,j})$ and $h_{i,j} = \mathcal{H}_i(M, \sigma_{0,j}, \ldots, \sigma_{i-1,j}, h_{1,j}, \ldots, h_{i-1,j})$ for $i = 2, \ldots, n$ and $j = 1, \ldots, r$. The value $\sigma_{i,j}$ depends on previous $\sigma_{0,j}, \ldots, \sigma_{i-1,j}$ and hash values $h_{1,j}, \ldots, h_{i,j}$. We require that the min-entropy of the random variable which outputs $\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_{n-1}$ must be in $\omega(|\mathcal{H}_n|)$.

Honest-Verifier Zero-Knowledge (HVZK) Let Test be an algorithm such that $\mathsf{Test}(\mathsf{pk}, \mathsf{sk}) = 1$ iff $(\mathsf{sk}, \mathsf{pk})$ belongs to the range of $\mathsf{KGen}(1^\kappa)$. Let PK (resp. SK) be a set of public (resp. matching secret) keys. The HVZK property states that there exists a PPT algorithm Z, the zero-knowledge simulator, controlling the random oracles, such that for any pair of PPT algorithms $D = (D_0, D_1)$ the following distributions are computationally indistinguishable:

- Let $(PK, SK, \mathsf{sk}, M, state) \leftarrow D_0(1^\kappa)$. If $\mathsf{Test}(\mathsf{pk}, \mathsf{sk}) = 1$, set

$$\sigma = (\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_n, \mathbf{h}_1, \ldots, \mathbf{h}_n) \leftarrow \mathsf{Sign}(\mathsf{pk}, \mathsf{sk}, PK, M) ;$$

  otherwise set $\sigma \leftarrow \perp$. Output $D_1(\sigma, state)$.

- Let $(PK, SK, \mathsf{sk}, M, state) \leftarrow D_0(1^\kappa)$. If $\mathsf{Test}(\mathsf{pk}, \mathsf{sk}) = 1$, set

$$\sigma = (\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_n, \mathbf{h}_1, \ldots, \mathbf{h}_n) \leftarrow Z(\mathsf{pk}, PK, M, 1) ;$$

  otherwise set $\sigma \leftarrow Z(\mathsf{pk}, PK, M, 0)$. Output $D_1(\sigma, state)$.

where PK (resp. SK) is a set of public (resp. secret) keys.

No-Message Attack Model.

The following lemma proves validity of the Extended Ring Forking Lemma in the no-message attack model for n-generic ring signature schemes, where the adversary has to forge a valid signature knowing only the verification keys.

Lemma 5.7. *Let* RS *be an n-generic ring signature scheme with security parameter κ, and let r be the number of ring members. Let $\mathcal{A}$ be a PPT algorithm given only the public data as input. If $\mathcal{A}$ can find a valid signature* $(\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_n, \mathbf{h}_1, \ldots, \mathbf{h}_n)$ *for*

a message M with a non-negligible probability, after asking the n random oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$ polynomially often (in κ), then, a replay of this machine with the same random tape, the same first oracles $\mathcal{O}_1, \ldots, \mathcal{O}_{n-1}$ and a different last oracle $\mathcal{O}_n$, outputs two valid signatures $(\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_n, \mathbf{h}_1, \ldots, \mathbf{h}_n)$ and $(\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_{n-1}, \boldsymbol{\sigma}'_n, \mathbf{h}_1, \ldots, \mathbf{h}_{n-1}, \mathbf{h}'_n)$ for the same message M with a non-negligible probability such that $\mathbf{h}_n \neq \mathbf{h}'_n$.

Proof. We are given a no-message adversary $\mathcal{A}$, which is a probabilistic polynomial time Turing machine with a random tape ω taken from a set R_ω. $\mathcal{A}$ may ask $q_1, \ldots, q_n$ (polynomially bounded) queries to random oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$ with $q_j^{(i)}$ denoting the j-query to oracle $\mathcal{O}_i$. Furthermore, let $r_i = (r_1^{(i)}, \ldots, r_{q_i}^{(i)})$ be the answers from oracle $\mathcal{O}_i$. Let S_i denotes the set of all possible answers from $\mathcal{O}_i$ for $1 \leq i \leq n$, i.e., $\{r_1^{(i)}, \ldots, r_{q_i}^{(i)}\} \in S_i$.

Note that $\boldsymbol{\sigma}_0$ is a tuple $\boldsymbol{\sigma}_0 = (\sigma_{0,1}, \ldots, \sigma_{0,r})$, and $\boldsymbol{\sigma}_i$ (resp. $\mathbf{h}_i$) is a tuple $\boldsymbol{\sigma}_i = (\sigma_{i,1}, \ldots, \sigma_{i,r})$ (resp. $\mathbf{h}_i = (h_{i,1}, \ldots, h_{i,r})$) for $1 \leq i \leq n$. For the sake of shorter formulas, let $\sigma_{i:k,j}$ (resp. $h_{i:k,j}$) denote the commitments (resp. challenges) $\sigma_{i,j}, \ldots, \sigma_{k,j}$ (resp. $h_{i,j}, \ldots, h_{k,j}$).

Let us denote by:

$\mathcal{E}$ the event that $\mathcal{A}$ can produce a valid signature $(\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_n, \mathbf{h}_1, \ldots \mathbf{h}_n)$ for a message M by using random tape ω and the answers r_i for $1 \leq i \leq n$. Note that a valid signature implies $h_{i,j} = \mathcal{O}_i(M, \sigma_{0,j}, \ldots, \sigma_{i-1,j}, h_{1,j}, \ldots, h_{i-1,j})$.

$\mathcal{F}$ the event that $\mathcal{A}$ has queried the oracle $\mathcal{O}_n$ with input $(M, \sigma_{0:n-1,j}, h_{1:n-1,j})$ for all ring members $j \leq r$, i.e.,

$$\forall j \leq r \; \exists \ell_j \leq q_n, \; q_{\ell_j}^{(n)} = (M, \sigma_{0:n-1,j}, h_{1:n-1,j}) \; .$$

Accordingly, its complement $\neg\mathcal{F}$ denotes

$$\exists j \leq r \; \forall \ell_j \leq q_n, q_{\ell_j}^{(n)} \neq (M, \sigma_{0:n-1,j}, h_{1:n-1,j}) \; .$$

By hypothesis of the lemma, the probability that event $\mathcal{E}$ occurs ($\Pr[\mathcal{E}]$), is non-negligible, i.e., there exists a polynomial function $T(\kappa)$ such that: $\Pr[\mathcal{E}] \geq \frac{1}{T(\kappa)}$. We know that

$$\Pr[\mathcal{E}] = \Pr[\mathcal{E} \wedge \mathcal{F}] + \Pr[\mathcal{E} \wedge \neg\mathcal{F}] . \tag{5.1}$$

Furthermore, we have

$$\begin{aligned}
&\Pr[\forall j \leq r : h_{n,j} = \mathcal{O}_n(M, \sigma_{0:n-1,j}, h_{1:n-1,j}) \wedge \neg\mathcal{F}] \\
&\quad = \Pr[\forall j \leq r : h_{n,j} = \mathcal{O}_n(M, \sigma_{0:n-1,j}, h_{1:n-1,j} \mid \neg\mathcal{F}] \cdot \Pr[\neg\mathcal{F}] \\
&\quad \leq \Pr[\forall j \leq r : h_{n,j} = \mathcal{O}_n(M, \sigma_{0:n-1,j}, h_{1:n-1,j} \mid \neg\mathcal{F}] \\
&\quad \leq \frac{1}{2^{r\kappa_n}},
\end{aligned}$$

because the output of $\mathcal{O}_n$ is unpredictable and $(M, \boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_{n-1})$ has a high min-entropy. The event $\mathcal{E}$ implies that $\forall j \leq r : h_{n,j} = \mathcal{O}_n(M, \sigma_{0:n-1,j}, h_{1:n-1,j})$, and thus we get

$$\begin{aligned} &\Pr[\mathcal{E} \wedge \neg\mathcal{F}] \\ &\leq \Pr[\forall j \leq r : h_{n,j} = \mathcal{O}_n(M, \sigma_{0:n-1,j}, h_{1:n-1,j}) \wedge \neg\mathcal{F}] \\ &\leq \frac{1}{2^{r\kappa_n}} \end{aligned} \tag{5.2}$$

Relations (5.1) and (5.2) lead to

$$\Pr[\mathcal{E} \wedge \ \mathcal{F}] \geq \frac{1}{T(\kappa)} - \frac{1}{2^{r\kappa_n}} \geq \frac{1}{T'(\kappa)}. \tag{5.3}$$

Note that a polynomial $T'(\cdot)$ must exist since the difference between a non-negligible and negligible term is non-negligible. Therefore, for all $j \leq r$, $\exists \ell_j \leq q_n$ such that

$$\Pr\left[\mathcal{E} \wedge q_{\ell_j}^{(n)} = (M, \sigma_{0:n-1,j}, h_{1:n-1,j})\right] \geq \frac{1}{rq_nT'(\kappa)}.$$

Notice, that in probabilistic term, the event associated to $\mathcal{F}$ is

$$\cap_{j=1}^{r} \cup_{\ell=1}^{q_n} F(\ell, j)$$

where $F(\ell, j)$ is the event $q_\ell^{(n)} = (M, \sigma_{0:n-1,j}, h_{1:n-1,j})$.

Suppose that $\exists j_0 \leqslant r$ such that $\forall \ell \in \{1, \ldots, q_n\}$,

$$\Pr\left[\mathcal{E} \wedge q_\ell^{(n)} = (M, \sigma_{0:n-1,j_0}, h_{1:n-1,j_0})\right] < \frac{1}{rq_nT'(\kappa)}. \tag{5.4}$$

Then

$$\Pr[\mathcal{E} \wedge \ \mathcal{F}] = \Pr\left[\mathcal{E} \wedge \ \cap_{j=1}^{r} \cup_{\ell=1}^{q_n} F(\ell, j)\right] \leq \Pr\left[\mathcal{E} \wedge \ \cup_{\ell=1}^{q_n} F(\ell, j_0)\right]$$

Now

$$\Pr\left[\mathcal{E} \wedge \ \cup_{\ell=1}^{q_n} F(\ell, j_0)\right] \leqslant \sum_{\ell=1}^{q_n} \Pr[\mathcal{E} \wedge \ F(\ell, j_0)]$$

and since from (5.4)

$$\Pr[\mathcal{E} \wedge \ F(\ell, j_0)] < \frac{1}{rq_nT'(\kappa)}$$

This leads to

$$\Pr[\mathcal{E} \wedge \ \mathcal{F}] < \frac{1}{rT'(\kappa)}$$

which is in contradiction with (5.3).

For each $j \in \{1, \ldots, r\}$, let $\ell_j \in \{1, \ldots, q_n\}$ be such that $q_{\ell_j}^{(n)} = (M, \sigma_{0:n-1,j}, h_{1:n-1,j})$. We denote by β_ℓ the maximum coordinates of ℓ, i.e. $\beta_\ell := \max_j\{\ell_j\}$. Then, we can apply the splitting lemma (see Lemma 4.3).

Let us now define

$$B = \{(\omega, r_1, \ldots, r_n) \text{ s.t. } \mathcal{E} \wedge q_{\beta_\ell}^{(n)} = (M, \sigma_{0:n-1,\ell_j}, h_{1:n-1,\ell_j})\}.$$

Since, $B \subset R_\omega \times S_1^{q_1} \times \ldots \times S_n^{q_n}$ and $\Pr[B] \geq \frac{1}{rq_nT'(\kappa)}$, by using the splitting lemma we have:

- $\exists \Omega \subset R_\omega$ such that $\Pr[\omega \in \Omega] \geq \frac{1}{2rq_nT'(\kappa)}$.
- $\forall \omega \in \Omega$, $\Pr\Big[(\omega, r_1^{(1)}, \ldots, r_{q_1}^{(1)}, \ldots, r_1^{(n)}, \ldots, r_{q_n}^{(n)}) \in B\Big] \geq \frac{1}{2rq_nT'(\kappa)}$, where the probability is taken over $S_1^{q_1} \times \ldots \times S_n^{q_n}$.

We define

$$B' = \{(\omega, r_1, \ldots, r_n) \text{ s.t. } (\omega, r_1, \ldots, r_n) \in B \wedge \omega \in \Omega\}.$$

Since, $B' \subset (R_\omega \times S_1^{q_1} \times \ldots \times S_n^{\beta_\ell - 1}) \times S_n^{q_n - \beta_\ell + 1}$, by using the splitting lemma again we get

- $\exists \Omega' \subset R_\omega \times S_1^{q_1} \times \ldots \times S_n^{\beta_\ell - 1}$ such that $\Pr\Big[(\omega, r_1, \ldots, r_{n-1}, r_1^{(n)}, \ldots, r_{\beta_\ell - 1}^{(n)}) \in \Omega'\Big] \geq \frac{1}{4rq_nT'(\kappa)}$.
- $\forall (\omega, r_1, \ldots, r_{n-1}, r_1^{(n)}, \ldots, r_{\beta_\ell - 1}^{(n)}) \in \Omega'$, $\Pr\Big[(\omega, r_1, \ldots, r_{n-1}, r_1^{(n)}, \ldots, r_{\beta_\ell - 1}^{(n)}, r_{\beta_\ell}^{(n)}, \ldots, r_{q_n}^{(n)}) \in B'\Big] \geq \frac{1}{4rq_nT'(\kappa)}$, where the probability is taken over $S_n^{q_n - \beta_\ell + 1}$.

As a result, if we choose β_ℓ, ω, $(r_1, \ldots, r_{n-1}, r_1^{(n)}, \ldots, r_{\beta_\ell - 1}^{(n)}), (r_{\beta_\ell}^{(n)}, \ldots, r_{q_n}^{(n)})$, and $(r_{\beta_\ell}^{\prime(n)}, \ldots, r_{q_n}^{\prime(n)})$ randomly, then we obtain with a non-negligible probability two valid signatures $(\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_n, \mathbf{h}_1, \ldots, \mathbf{h}_n)$ and $(\boldsymbol{\sigma}_0, \ldots, , \boldsymbol{\sigma}_{n-1}, \boldsymbol{\sigma}'_n, \mathbf{h}_1, \ldots, \mathbf{h}'_n)$ for the same message M such that $\mathbf{h}_n \neq \mathbf{h}'_n$. □

Adaptively Chosen-Message Attacks.

So far, we considered the security of n-generic ring signature schemes against no-message attacks. However, ring signatures require to satisfy security against adaptively chosen-message attacks to achieve the standard security level.

Informally, chosen-message attacks work as follows. After an adversary receives the public key of the ring signature scheme, he may ask queries to the signing oracle which expects a message M and a party identifier *pid* as input and outputs a

signature σ. At some point the adversary outputs a message M^* and a signature σ^*. If M^* was not queried before and the signature σ^* is valid, we declare the adversary as successful.

In case we prove an n-generic ring signature scheme to be unforgeable (against adaptively chosen-message attacks) in the random oracle model, the adversary may query, in addition to the signing oracle, also the random oracle (polynomially often in the security parameter).

The following theorem states the Extended Ring Forking Lemma against adaptive chosen-message attacks.

Theorem 5.8 (The Chosen-Message Extended Ring Forking Lemma)**.** *Let* RS *be an n-generic ring signature scheme with security parameter κ, and let r be the number of ring members. Let $\mathcal{A}$ be a PPT algorithm given only the public data as input. If $\mathcal{A}$ can find a valid signature $(\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_n, \mathbf{h}_1, \ldots, \mathbf{h}_n)$ for a message M with a non-negligible probability, after asking the n random oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$ and some real signer of the ring polynomially often (in κ), then, a replay of this machine with the same random tape, the same first oracles $\mathcal{O}_1, \ldots, \mathcal{O}_{n-1}$ and a different last oracle $\mathcal{O}_n$, outputs two valid signatures $(\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_n, \mathbf{h}_1, \ldots, \mathbf{h}_n)$ and $(\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_{n-1}, \boldsymbol{\sigma}'_n, \mathbf{h}_1, \ldots, \mathbf{h}_{n-1}, \mathbf{h}'_n)$ for the same message M with a non-negligible probability such that $\mathbf{h}_n \neq \mathbf{h}'_n$.*

Proof. We consider a PPT algorithm $\mathcal{B}$ that executes $\mathcal{A}$ in such a way that $\mathcal{B}$ simulates the environment of $\mathcal{A}$. Therefore, $\mathcal{B}$ must simulate the interactions of $\mathcal{A}$ with random oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$ and with real signers. Then we could see $\mathcal{B}$ as an algorithm performing a no-message attack against the ring signature scheme RS.

We denote by $q_1^{(i)}, \ldots, q_{q_i}^{(i)}$ the q_i distinct queries to random oracle $\mathcal{O}_i$, and by $M^{(1)}, \ldots, M^{(q_s)}$ the q_s queries (possibly repeated) to the real signers. Note that q_i's and q_s are polynomially bounded in κ. Since RS is an n-generic ring signature scheme, RS satisfies the honest-verifier zero knowledge property and thus there exists an efficient simulator which outputs a valid ring signature for an adversary chosen message without the knowledge of the secret key and which output distribution is indistinguishable from a real signer. Using this simulator, $\mathcal{B}$ can perfectly simulate the answers of the real ring of signers. For a message $M^{(l)}$, the simulator answers a tuple $(M^{(l)}, \boldsymbol{\sigma}_0^{(l)}, \ldots, \boldsymbol{\sigma}_n^{(l)}, \mathbf{h}_1^{(l)}, \ldots, \mathbf{h}_n^{(l)})$ where $\boldsymbol{\sigma}_0^{(l)} = (\sigma_{0,1}^{(l)}, \ldots, \sigma_{0,r}^{(l)})$, $\boldsymbol{\sigma}_i^{(l)} = (\sigma_{i,1}^{(l)}, \ldots, \sigma_{i,r}^{(l)})$ and $\mathbf{h}_i^{(l)} = (h_{i,1}^{(l)}, \ldots, h_{i,r}^{(l)})$ for $i = 1, \ldots, n$. Then $\mathcal{B}$ constructs random oracles $\mathcal{O}_1^{\mathcal{B}}, \ldots, \mathcal{O}_n^{\mathcal{B}}$ by storing in "random oracle lists" the relations $\mathcal{O}_i^{\mathcal{B}}(M^{(l)}, \sigma_{1,j}^{(l)}, \ldots, \sigma_{i-1,j}^{(l)}, h_{1,j}^{(l)}, \ldots, h_{i-1,j}^{(l)}) = h_{i,j}^{(l)}$ for all $j \in \{1, \ldots, r\}$, $i \in \{1, \ldots, n\}$ and $l \in \{1, \ldots, q_s\}$. When $\mathcal{A}$ makes a query $q_j^{(i)}$ to the random oracle $\mathcal{O}_i$, $\mathcal{B}$ looks in the random oracle list of $\mathcal{O}_i^{\mathcal{B}}$. If this value is in the list, $\mathcal{B}$ outputs the corresponding $\mathcal{O}_i^{\mathcal{B}}(q_j^{(i)})$. Otherwise, $\mathcal{B}$ chooses a random value h, sends it to $\mathcal{A}$ and stores the relation $\mathcal{O}_i^{\mathcal{B}}(q_j^{(i)}) = h$.

Now we need to consider potential "collisions" of queries in the random oracles. If the simulator outputs ring signatures which are indistinguishable from the ones

produced by a real signer of the ring, then we have that no $\sigma_{i,j}^{(l)}$ can appear with probability greater than $1/2^\kappa$ in a simulated ring signature, too. Since the values $h_{i,j}^{(l)}$ are outputs of the random oracle, we have that a determined $h_{i,j}^{(l)}$ appears in a ring signature (real or simulated) with probability less than $1/2^\kappa$.

Then, three kinds of collision can occur:

- A tuple $(M^{(l)}, \sigma_{0,j}^{(l)}, \ldots, \sigma_{i-1,j}^{(l)}, h_{1,j}^{(l)}, \ldots, h_{i-1,j}^{(l)})$ that the simulator outputs, as part of a simulated ring signature, has been asked before to the random oracle $\mathcal{O}_i$ by the adversary. In this case, it is quite unlikely that the relation $\mathcal{O}_i^{\mathcal{B}}(M^{(l)}, \sigma_{0,j}^{(l)}, \ldots, \sigma_{i-1,j}^{(l)}, h_{1,j}^{(l)}, \ldots, h_{i-1,j}^{(l)}) = h_{i,j}^{(l)}$ corresponding to the values output by the simulator coincides with the relation previously stored in the random oracle lists. The probability of such a collision in all oracles is, however, less than $\frac{(q_1+\ldots+q_n)rq_s}{2^\kappa}$.

- A tuple $(M^{(l_1)}, \sigma_{0,j_1}^{(l_1)}, \ldots, \sigma_{i-1,j_1}^{(l_1)}, h_{1,j_1}^{(l_1)}, \ldots, h_{i-1,j_1}^{(l_1)})$ that the simulator outputs, as part of a simulated ring signature, is exactly equal to another tuple $(M^{(l_2)}, \sigma_{0,j_2}^{(l_2)}, \ldots, \sigma_{i-1,j_2}^{(l_2)}, h_{1,j_2}^{(l_2)}, \ldots, h_{i-1,j_2}^{(l_2)})$ also output by the simulator. The probability of this collision is less than $\frac{(rq_s)^2}{2} \cdot \frac{1}{2^\kappa}$.

- Two answers h_{i,j_1} and h_{i,j_2} of the random oracle $\mathcal{O}_j$ chosen at random by $\mathcal{B}$ are exactly equal, while the two corresponding inputs $(M^{(1)}, \sigma_{0,j_1}^{(1)}, \ldots, \sigma_{i-1,j_1}^{(1)}, h_{1,j_1}^{(1)}, \ldots, h_{i-1,j_1}^{(1)})$ and $(M^{(2)}, \sigma_{0,j_2}^{(2)}, \ldots, \sigma_{i-1,j_2}^{(2)}, h_{1,j_2}^{(2)}, \ldots, h_{i-1,j_2}^{(2)})$ are different. The probability of such an event for all random oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$ is less than $\frac{((q_1+\ldots+q_n)+rq_s)^2}{2} \cdot \frac{1}{2^\kappa}$.

Let $q := q_1 + \ldots + q_n$. Now we can compute:

$$\begin{aligned}
\Pr_{(\omega,\mathcal{O}_1,\ldots,\mathcal{O}_n)}[\mathcal{B} \text{ succeeds}] &= \Pr_{(\omega,\mathcal{O}_1,\ldots,\mathcal{O}_n)}[\text{no-collisions and } \mathcal{A} \text{ succeeds}] \\
&\geq \Pr_{(\omega,\mathcal{O}_1,\ldots,\mathcal{O}_n)}[\mathcal{A} \text{ succeeds} \mid \text{no-collisions}] \\
&\quad -\Pr_{(\omega,\mathcal{O}_1,\ldots,\mathcal{O}_n)}[\text{no-collisions}] \\
&\geq \Pr_{(\omega,\mathcal{O}_1,\ldots,\mathcal{O}_n)}[\mathcal{A} \text{ succeeds}] \\
&\quad -\frac{qrq_s + (rq_s)^2 + 2(q + rq_s)}{2^{\kappa+1}}
\end{aligned}$$

The resulting term is non-negligible since $\Pr_{(\omega,\mathcal{O}_1,\ldots,\mathcal{O}_n)}[\mathcal{A} \text{ succeeds}]$ is assumed to be non-negligible and the second term is negligible in security parameter κ.

Summing up, we have an algorithm $\mathcal{B}$ that performs a no-message attack against the ring signature scheme RS in polynomial time with non-negligible probability of success. So we can use Lemma 5.7 applied to algorithm $\mathcal{B}$, and we will obtain two valid signatures again in polynomial time. □

As application of theorem 5.8 and using a reduction technique to the underlying hard problem, the security of ring signature schemes is guaranteed in the random-oracle model. The following theorem states this result in particular for threshold ring signature schemes derived from an ID scheme.

Theorem 5.9. *The resulting threshold ring signature scheme obtained from the CVE scheme is unforgeable under chosen-message attacks in the random-oracle model.*

Theorem 5.10. *Our threshold ring signature scheme obtained from the CVE scheme is anonymous in the random-oracle model.*

Proof. The second property we would like to examine is the anonymity property in the random-oracle model of the resulting threshold ring signature scheme obtained from the CVE scheme. In other words, the verifier must not be able to determine the identity of the real signers, a part from the fact that they were at least t among the n specified ring members. For the challenge 0 the response of both real signers and the non-signers are completely indistinguishable, since Θ, Σ_i, and γ_i are chosen uniformly at random and therefore the response is random. So the only possibility to identify non-signers is challenge 1. In this case the verifier receives a permuted value of the secret key without having access to the used permutation. As consequence, the anonymity of the signers is preserved.

The two last theorems permit us to conclude the security proof of the obtained threshold ring signature scheme. □

5.4.4 Performance aspect and comparison

In general the signature length of signature schemes derived from ID schemes is constrained by the number of rounds. Our proposal is built by applying the CVE scheme, which needs a smaller number of rounds to reach the same cheating probability as by Stern's ID scheme. For a cheating probability of 2^{-80}, one needs about 140 rounds for Aguilar et al.'s scheme and only 80 rounds for our proposal. This fact has a positive effect in terms of signature length for our proposal.

A second code-based threshold ring signature was proposed by Dallot and Vergnaud [28], this scheme uses the CFS signature scheme as basis, therefore it inherits the advantage to provide a shorter signature length, however it suffers from slow signature generation cost and large public key sizes.

Taking into account the performances of the ISD algorithm, we suggest the following parameters to reach 80 bit security.

For our scheme we take $q = 256$, $n = 128$, $r = 64$, and $\omega = 49$. For Aguilar et al.'s scheme, we need to take: $q = 2$, $n = 694$, $r = 347$, $\omega = 69$ [3]. For Dallot and Vergnaud's scheme we take $q = 2$, $n = 2^{22}$, $r = 198$, $\omega = 9$.

We considered that all seeds used are 128 bits long, the hash outputs are 160 bits long and the cheating probability is bounded by 2^{-80}. Table 5.1 presents a comparison between all existing code-based threshold ring signature schemes in terms of key sizes, signature length and the signing cost for the parameter set $(N, t) = (100, 50)$.

Recently, two post-quantum threshold ring signature schemes have been proposed which are related to our construction. The first one (CLRS) was proposed by Cayrel et al. presented in [20] and based on the hardness of SIS lattice problem. The second one (PBB) was proposed by Petzoldt et al. based on the MQ-problem of solving systems of quadratic equations over finite fields.

TRSS	Public key (KB)	Signature size (KB)	Signature cost (ops.)
Aguilar et. al's scheme	1470	2448	2^{30}
Dallot and Vergnaud's scheme	10137122	7	2^{35}
Our scheme	400	1946	2^{26}

Table 5.1: Comparison code-based threshold ring signature schemes

TRSS	Public key (KB)	Signature size (KB)	Area
CLRS	7168	14336	Lattices
PBB	3584	655	Multivariates
Our scheme	400	1946	Codes

Table 5.2: Comparison post-quantum threshold ring signature schemes

Table 5.2 compares our scheme with these two post-quantum threshold signature schemes for 80-bit security.

Remark 5.11. To further reduce the public key size, we can replace a random matrix H by a quasi-cyclic matrix respectively a quasi-dyadic matrix. In this case, we obtain a public key size in 12.5 KB for our construction, 8.47 KB for Aguilar et al.'s one.

5.5 Implementation Results

General remarks

The following tables show the timings we have obtained for a C implementation of our threshold ring identification scheme and Aguilar et al.'s one. The test system was an Intel(R) Core(TM)2 Duo CPU E8400@3.00GHz, running Debian 6.0.3. The sources have been compiled using gcc 4.6.2.

In all cases, we used parity check matrices in systematic form. Due to the row-major order of C, the transposed matrices have been stored. The tables show the setup time and the time running the protocol, where the setup time is consumed for the generation of the necessary public and private keys.

Finally, the use of quasi-dyadic matrices does not allow for all theoretically possible parameters. For instance, dyadic matrices have the dimension $2^p \times 2^p$ ($p \in \mathbb{N}$), which means that for quasi-dyadic matrices $r = b2^p$ for some $b \in \mathbb{N}$. In order to have comparable results and a uniform implementation, we have used this restriction for the random and the quasi-cyclic case as well.

Aguilar et. al's scheme

The number of rounds for the scheme has been set to 28 (cheating probability of 2^{-16}), the dimension of the parity check matrix H^T over $\mathbb{F}_2$ has been set to 704×352, but only the redundancy part has been stored in memory, which is of dimension 352×352 bits. The weight of the secrets has been set to 76. Table 5.3 shows the timing results.

Matrix Type	**Dim.** $[n \times r]$	**Weight**	**Setup** $[ms]$	**Protocol** $[ms]$	**Total** $[ms]$	**Sec.**$_{[bits]}$
Random	704×352	76	108.539	98.662	207.200	80
Quasi-dyadic	704×352	76	811.202	474.737	1285.939	80
Quasi-cyclic	704×352	76	476.796	302.935	779.731	80

Table 5.3: Timings for Aguilar et al.'s scheme.

Our proposal

Matrix Type	**Dim.** $[n \times r]$	**Weight**	**Setup** $[ms]$	**Protocol** $[ms]$	**Total** $[ms]$	**Sec.**$_{[bits]}$
Random	144×72	54	32.979	18.499	51.477	80
Quasi-dyadic	144×72	54	44.331	29.109	73.439	80
Quasi-cyclic	144×72	54	38.747	26.550	65.298	80

Table 5.4: Timings for our proposal.

For our scheme the parity check matrices H^T have been chosen over $\mathbb{F}_{2^8}$, mainly because in this case a field element fits exactly in one byte. The number of rounds has been set to 16 (cheating probability of 2^{-16}), the weight of the secrets has been set to 54. Table 5.4 shows the timing results.

Remark 5.12. The given implementation is given as a proof of concept. For instance, the communication between the leader and the signers takes place on the same machine, even inside the same executable. In reality, the signers would be located on different computers, having a different architecture, connected to the leader via network connections and the like. In such a heterogeneous scenario, the communication latency for those network connections had to be taken into account. It also might be possible that some signers use a very fast machine, whereas others use a very slow one. The interaction process would be dominated then by the slowest possible signer.

Resulting threshold ring signature scheme

In Table 5.5 we give some timings for the resulting signature scheme using the generalized Fiat-Shamir paradigm. We used the same settings as above, but run

Doc. $[MB]$	**Sig.** $[MB]$	**Dim.** $[n \times r]$	**Weight**	**Signing** $[ms]$	**Verification** $[ms]$	**Sec.**$_{[bits]}$
1	4	144×72	54	544	454	80
10	13	144×72	54	3643	3551	80
25	28	144×72	54	8803	8700	80

Table 5.5: Timings for our proposal.

the protocol with random matrices only. The savings using other matrix types is negligible compared to the gained signature sizes.
The signature sizes are not fixed, but show a small variation depending on the values chosen during the challenge step. More specifically, the answers transmitted for the cases b=0,1 vary in size, which effectively leads to varying signature sizes as well. The values are therefore average values obtained while running the protocol 80 rounds (cheating probability of 2^{-80}).

6 Conclusion and Future Work

The security of most public-key cryptosystems used in practice today will be threatened for the time when potential quantum computer become a reality. Besides the fact that code-based cryptography is believed to resist to quantum attacks, code-based cryptosystems are one of the most promising candidates of future cryptography mostly due to their simplicity of operations and high speed performance. However, if we consider, in particular, the question of designing code-based identification and signature schemes, all current proposals are rather impractical for many applications with constraint devices, e.g., smartcards. Beside the problem of public-key size, this is due to the very large communication complexity for identification scheme, and slow signing algorithms or security issues for signature schemes. In order to address these drawbacks, we have introduced in this thesis an improved five-pass zero-knowledge identification scheme whose security is directly derived from the hardness of decoding random codes. Our scheme has better communication complexity and smaller public key size compared to all code-based identification using random codes. We have also presented an extension of the Fiat-Shamir paradigm and the well-known Forking lemma, which can be used to construct efficient signatures from identification schemes with more than 3-pass. This was one open task for many recent works such as in [73]. We have applied our new framework in order to build signature from our proposed identification scheme. The implementation results has showed that the resulting scheme is very fast with signing/verification time around 2 ms. However, the size of the signature and the public key are still large (19KB) compared to classical schemes like RSA and DSA (< 2 KB). Our final result has concerned developing schemes with additional properties. To this end, we have applied our extended Fiat-Shamir to our proposed identification scheme in order to construct a threshold ring signature scheme, which is fully anonymous and unforgeable based on a proof of knowledge in the random oracle model. This proposal achieves a scheme with shorter signature length, smaller public key size and signature cost compared to schemes based on codes but still inefficient in practice. We conclude that a lot of effort is needed to have secure and practical code-based signature schemes which can be serious alternatives to the currently used cryptosystems. For example, it would be very helpful to see whether it is possible to modify existing ID schemes in order to obtain new ID protocols with negligible soundness error which would reduce the number of rounds to reach a perfect completeness.

References

[1] Abdalla, M., An, J. H., Bellare, M., Namprempre, C.: From identification to signatures via the Fiat-Shamir transform: Minimizing assumptions for security and forward-security. EUROCRYPT 2002, LNCS 2332, pages 418-433, Springer, 2002.

[2] Aguilar Melchor, C., Gaborit, P., Schrek, J.: A new zero-knowledge code based identification scheme with reduced communication. CoRR abs/1111.1644, 2011.

[3] Aguilar Melchor, C., Cayrel, P.-L., Gaborit, P.: A new efficient threshold ring signature scheme based on coding theory. PQCrypto 2008, LNCS 5299, pages 1-16, Springer, 2008.

[4] Aguilar Melchor, C., Cayrel, P.-L., Gaborit, P., Laguillaumie, F.: A New Efficient Threshold Ring Signature Scheme based on Coding Theory. Information Theory, IEEE Transactions on, 57(7):4833-4842, 2011.

[5] Augot, D., Finiasz, M.,Sendrier, N.: A Family of Fast Syndrome Based Cryptographic Hash Functions. In E. Dawson and S. Vaudenay (Eds.), MyCrypt 2005, LNCS 3615, pages 64-83. Springer, 2005.

[6] Barg, S.: Some New NP-Complete Coding Problems. Journal Probl. Peredachi Inf., 30(3):23-28, 1994.

[7] Barreto, P. S. L. M., Cayrel, P.-L., Misoczki, R., Niebuhr, R.: Quasi-dyadic CFS signatures. In Inscrypt 2010, LNCS 6584, pages 336-349, Springer, 2010.

[8] Barreto, P. S. L. M., Misoczki, R., Simplicio Jr., M. A.: One-time signature scheme from syndrome decoding over generic error-correcting codes. Journal of Systems and Software 84(2), 198-204, 2011.

[9] Becker, A., Joux, A., May, A., Thomae, E.: Decoding Random Binary Linear Codes in $2^{(n/20)}$: How 1+1=0 improves Information Set Decoding. Eurocrypt 2012, LNCS 7237, pages 520-536, Springer, 2012.

[10] Berlekamp, E., McEliece, R., van Tilborg, H.: On the inherent intractability of certain coding problems. IEEE Transactions on Information Theory, 24(3):384-386, 1978.

[11] Berger, T. P., Cayrel, P.-L., Gaborit, P., Otmani, O.: Reducing Key Length of the McEliece Cryptosystem. Africacrypt 2009, LNCS 5580, pages 77-97, Springer, 2009.

[12] Bernstein, D. J., Lange, T., Peters, C.: Smaller decoding exponents: ball-collision decoding. Crypto 2011, LNCS 6841, pages 743-760, Springer, 2011.

[13] Bernstein, D. J., Buchmann, J., Dahmen, E. (Eds.): Post-Quantum Cryptography. Springer, 2009, ISBN: 978-3-540-88701-0.

[14] Biswas, B., Sendrier, N.: Mceliece Cryptosystem Implementation: Theory and Practice. In PQCrypto 2008, LNCS 5299, pages 47-62, Springer, 2008.

[15] Bresson, E., Stern, J., Szydlo, M.: Threshold Ring Signatures and Applications to Ad-hoc Groups. CRYPTO 2002, LNCS 2442, pages 465-480, Springer, 2002.

[16] Cayrel, P.-L., Meziani, M.: Post-quantum Cryptography: Code-Based Signatures. ISA 2010, LNCS 6059, pages 82-99, Springer, 2010.

[17] Cayrel, P.-L., El Yousfi Alaoui, S. M., Véron, P., Hoffmann, G.: An improved threshold ring signature scheme based on error correcting codes. International Workshop on the Arithmetic of Finite Fields, WAIFI 2012, LNCS 7374, pages 54-63, Springer, 2012.

[18] Cayrel, P.-L., Lindner, R., Rückert, M., Silva, R.: Improved zero-knowledge identification with lattices. ProvSec 2010, LNCS 6212, pages 1-17, Springer, 2010.

[19] Cayrel, P.-L., Véron, P., El Yousfi Alaoui, S. M.: A new identification scheme based on syndrome decoding. SAC 2010, LNCS 6544, pages 170-186, Springer, 2010.

[20] Cayrel, P.-L., Lindner, R., Rückert, M., Silva, R.: A Lattice-Based Threshold Ring Signature Scheme. LATINCRYPT 2010, LNCS 6216, pages 255-272, Springer, 2010.

[21] Cayrel, P.-L., Otmani, A., Vergnaud, D.: On Kabatianskii-Krouk-Smeets Signatures. In Carlet, C., Sunar, B. (eds.) WAIFI 2007, LNCS 4547, pages 237-251, Springer, 2007.

[22] Cayrel, P.-L., Gaborit, P., Girault, M.: Identity-based identification and signature schemes using correcting codes. PQCrypto 2008, International Workshop on Coding and Cryptography, WCC 2007, pages 69-78, Springer, 2008.

[23] Cayrel, P.-L., Gaborit, P., Prouff, E.: Secure Implementation of the Stern Authentication and Signature Schemes for Low-Resource Devices. CARDIS 2008, LNCS 5189, pages 191-205, Springer, 2008.

[24] Chen, K.: Improved Girault identification scheme. Journal of Electronics Letters, IEE., 30(19):1590-1591, 1994.

[25] C. Yao, A., Zhao, Y.:. Digital Signatures from Challenge-Divided Sigma-Protocols. IACR Cryptology ePrint Archive, Report 2012/001, 2012.

[26] Courtois, N., Finiasz, M., Sendrier, N.: How to Achieve a McEliece-based Digital Signature Scheme. Asiacrypt 2001, LNCS 2248, pages 157-174, Springer, 2001.

[27] Dallot, L.: Towards a Concrete Security Proof of Courtois, Finiasz and Sendrier Signature Scheme. In WEWoRC 2007, LNCS 4945, pages 65-77, Springer, 2007.

[28] Dallot, L., Dallot, D.: Provably Secure Code-Based Threshold Ring Signatures. Proceedings of the 12th IMA International Conference on Cryptography and Coding 2009, LNCS 5921, pages 222-235, Springer, 2009.

[29] El Yousfi Alaoui, S. M., Özgür, D., Véron, P., Galindo, D., Cayrel, P.-L.: Extended Security Arguments for Signature Schemes. The 5th African International Conference on Cryptology, AfricaCrypt 2012, LNCS 7374, pages 19-34, Springer, 2012.

[30] Faugère, J.-C., Otmani, A., Perret, L., Tillich, J.-P.: Algebraic Cryptanalysis of McEliece Variants with Compact Keys. EUROCRYPT 2010, LNCS 6110, pages 279-298, Springer, 2010.

[31] Faugère, J.-C., Gauthier, V., Otmani, A., Perret, L., Tillich, J.-P.: A Distinguisher for High Rate McEliece Cryptosystems. eprint Report 2010/331, 2010.

[32] Fiat, A., Shamir, F.: How to prove yourself: practical solutions to identification and signature problems. CRYPTO 1986, LNCS 263, pages 186-194, Springer, 1986.

[33] Finiasz, M.: Parallel-CFS: Strengthening the CFS Mc-Eliece-based Signature Scheme. In Selected Areas in Cryptography 2010, LNCS 6544, pages 159-170. Springer, 2010.

[34] Finiasz, M., Sendrier, N.: Security Bounds for the Design of Code-based Cryptosystems. In Asiacrypt 2009, LNCS 5912, pages 88-105, Springer, 2009.

[35] Gaborit, P., Girault, M.: Lightweight code-based authentication and signature. IEEE International Symposium on Information Theory – ISIT'2007,:191-195, 2007.

[36] Gaborit, P., Zémor, G.: Asymptotic improvement of the GilbertVarshamov bound for linear codes, in Proceeedings of ISIT06, pages 287-291, 2006.

[37] Gaborit, P., Schrek, J.: Efficient code-based one-time signature from automorphism groups with syndrome compatibility. IEEE International Symposium on Information Theory – ISIT'2012, pages 1982-1986, Springer, 2012.

[38] Gaborit, P., Laudauroux, C., Sendrier, N.: SYND: A Fast Code-Based Stream Cipher with a Security Reduction. In Proceeedings of ISIT'07, 2007.

[39] Gauthier Umana, V., Leander, G.: Practical Key Recovery Attacks On Two McEliece Variants. IACR Cryptology ePrint Archive, Report 2009/509, 2009.

[40] Gilbert, E. N.: A comparison of signalling alphabets. The Bell system technical journal, 31:504-522, 1952.

[41] Girault, P., Poupard, G., Stern, J.: On the Fly Authentication and Signature Schemes based on Groups of Unknown Order. J. Cryptology, 4(3):463-487, 2006.

[42] Girault, M.: A (non-practical) three-pass identification protocol using coding theory. Advances in Cryptology - AUSCRYPT 1990, LNCS 453, pages 265-272, Springer, 1990.

[43] Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof-systems. STOC 1985, acmid 22178, pages 291-304, ACM, 1985.

[44] Goldreich, O.: Zero-knowledge twenty years after its invention. IACR Cryptology ePrint Archive, Report 2002/186, 2002.

[45] Guillou, L. C., Quisquater, J. J.: A paradoxical identity-based signature scheme resulting from zero-knowledge. CRYPTO 1988, LNCS 403, pages 216-231, Springer, 1990.

[46] Harari, J.: A new authentication algorithm. 3rd International Colloquium on Coding Theory and Applications, pages 91-105, Springer, 1989.

[47] Herranz, J., Sáez, G.: Forking Lemmas for Ring Signature Schemes. INDOCRYPT 2003, LNCS 2904, pages 266-279, Springer, 2003.

[48] Jaulmes, É., Joux, A.: Cryptanalysis of PKP: a new approach. PKC 2001, LNCS 1992, pages 165-172, Springer, 2001.

[49] Kabatianskii, G., Krouk, E., Smeets, B.: A Digital Signature Scheme based on Random Error-Correcting Codes. In: Darnell, M.J. (ed.) Cryptography and Coding 1997, LNCS 1355, pages 161-167, Springer, 1997.

[50] May, A., Meurer, A., Thomae, E.: Decoding Random Linear Codes in $\mathcal{O}(2^{0.054n})$. Asiacrypt 2011, LNCS 7073, pages 107-124, Springer, 2011.

[51] MacWilliams, F.J., Sloane, N.J.A.: The Theory of Error-Correcting Codes. Elsevier/North Holland, Amsterdam, 1977. ISBN 0444851933.

[52] McEliece, R.: A Public-Key Cryptosystem based on Algebraic Coding Theory. The Deep Space Network Progress Report, DSN PR, pages 114-116, 1978.

[53] Misoczki, R., Barreto, P. S. L. M.: Compact McEliece Keys from Goppa Codes. SAC 2009, LNCS 5867, pages 376-392, Springer, 2009.

[54] National Institute of Standards and Technology: Digital signature standard (DSS), FIPS PUB 186-3, 2006. Available at `http://csrc.nist.gov/publications/fips/`.

[55] Niebuhr, R., Cayrel, P.-L., Bulygin, S., Buchmann, J.: On lower bounds for Information Set Decoding over Fq. SCC 2010, RHUL, London, UK, pages 143-157, Springer, 2001.

[56] Niebuhr, R., Cayrel, P.-L., Bulygin, S., Buchmann, J.: On lower bounds for Information Set Decoding over Fq. Proceedings of the 2nd International Conference on Symbolic Computation and Cryptography - SCC, pages 143-157, Springer, 2010.

[57] Niederreiter, R.: Knapsack-type Cryptosystems and Algebraic Coding Theory. Problems of Control and Information Theory, 15(2):159-166, 1986.

[58] Ohta, K., Okamoto, T.:. On Concrete Security Treatment of Signatures Derived from Identification. CRYPTO 1998, LNCS 1462, pages 354-369, Springer, 1998.

[59] Okamoto, T.: Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes. CRYPTO 1992, LNCS 740, pages 31-53, Springer, 1993.

[60] Otmani, A., Tillich, J.-P.: An Efficient Attack on All Concrete KKS Proposals. In B.-Y. Yang, editor, PQCrypto, LNCS 7071, pages 98-116. Springer, 2011.

[61] Overbeck, R.: A Step Towards QC Blind Signatures. IACR Cryptology ePrint Archive 2009: 102 (2009).

[62] Overbeck, R., Sendrier, N.: Code-based cryptography. In D. J. Bernstein, J. Buchmann, and E. Dahmen, editors, Post-Quantum Cryptography, pages 95-145. Springer Berlin Heidelberg, 2009.

[63] Peters, C.: Information-Set Decoding for Linear Codes over F_q. PQCrypto 2010, LNCS 6061, pages 81-94, Springer, 2010.

[64] Pierce, J. N.: Limit distributions of the minimum distance of random linear codes. IEEE Trans. Inf. theory, 13(4):595-599, 1967.

[65] Pointcheval, D., Stern, J.: Security proofs for signature schemes. EUROCRYPT 1996, LNCS 1070, pages 387-398, Springer, 1996.

[66] Pointcheval, D.: A new identification scheme based on the perceptrons problem. EUROCRYPT 1995, LNCS 921, pages 319-328, Springer, 1995.

[67] Pointcheval, D., Poupart, G.: A New NP-Complete Problem and Public-Key Identification. Journal Des. Codes Cryptography, 28(1):5-31, 2003.

[68] Pointcheval, D., Stern, J.: Security Arguments for Digital Signatures and Blind Signatures. J. Cryptology, 13(3):361-396, 2000.

[69] Poupard, G.:. A Realistic Security Analysis of Identification Schemes based on Combinatorial Problems. European Transactions on Telecommunications, 8(5):471-480, 1997.

[70] Preetha Mathew, P., Vasant, S., Rangan, CP.: On Provably Secure Code-based Signature and Signcryption Scheme. eprint Report 2012/585, 2012.

[71] Rivest, R., Shamir, A., and Adleman, L.: A Method for obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM, 21(2):120-126, 1978.

[72] Rivest, R., Shamir, L., Tauman, Y.: How to Leak a Secret: Theory and Applications of Ring Signatures. Essays in Memory of Shimon Even, pages 164-186, 2010.

[73] Sakumoto, K., Shirai, T., Hiwatari, H.: Public-Key Identification Schemes Based on Multivariate Quadratic Polynomials. CRYPTO 2011, LNCS 6841, pages 706-723, Springer, 2011.

[74] Schnorr, C. P.: Efficient Identification and Signatures for Smart Cards. Crypto 1989, LNCS 435, pages 239-52, Springer, 1989.

[75] Sendrier, N.: Finding the permutation between equivalent linear codes: The support splitting algorithm, IEEE Trans. Inform. Theory 46(4): 1193-1203, 2000.

[76] Shamir, A.: An Efficient Identification Scheme based on Permuted Kernels (Extended Abstract). CRYPTO 1989, LNCS 435, pages 606-609, Springer, 1989.

[77] Shannon, C. E.: A mathematical theory of communication. Bell System Technical Journal, 27:379-423, 1948.

[78] Shor, P. W.: Algorithms for Quantum Computation: Discrete Logarithms and Factoring. Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science - FOCS 1994, pages 124-134, IEEE Computer Society Press, 1994.

[79] Shoup, V.: On the Security of a Practical Identification Scheme. J. Cryptology, 12(4):247-260, 1999.

[80] Silva, R., Cayrel, P.-L., Lindner, R.: Zero-knowledge Identification based on Lattices with Low Communication Costs. XI Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais, 8:95-107, 2011.

[81] Stern, J.: A new identification scheme based on syndrome decoding. CRYPTO 93, LNCS 773, pages 13-21, Springer, 1993.

[82] Stern, J.: Designing Identification Schemes with Keys of Short Size. CRYPTO 1994, LNCS 839, pages 164-173, Springer, 1994.

[83] Stern, J.: A new paradigm for public key identification. IEEE Transactions on Information theory 42, 1757-1768, 1996.

[84] Stern, J.: An Alternative to the Fiat-Shamir Protocol. EUROCRYPT 1989, LNCS 434, pages 173-180, Springer, 1989.

[85] Valery D. G.: Rational representation of codes and (L, g)-codes. Problemy Peredachi Informatsii, 7(3):41-49, 1971.

[86] Valery D. G.: Codes associated with divisors. Problemy Peredachi Informatsii, 13(1):33-39, 1977.

[87] Varshamov, R. R.: Estimate of the number of signals in error-correcting codes. Dokl. Acad. Nauk SSSR, 117:739-741, 1957.

[88] Véron, P.: Improved Identification Schemes Based on Error-Correcting Codes. Applicable Algebra in Engineering, Communication and Computing, 8(1):57-69, 1996.

[89] Véron, P.: Cryptanalysis of Hararis identification scheme, in Cryptography and Coding, 5th IMA Conference, LNCS 1025, pages 264-269, Springer, 1995.

[90] Wolf, C., Preneel, B.: $\mathcal{MQ}^*$-IP: An Identity-based Identification Scheme without Number-theoric Assumptions. IACR Cryptology ePrint Archive, Report 2010/087, 2010.

[91] Zheng, D., Li, X., Chen, K.: Code-based Ring Signature Scheme. I. J. Network Security 5(2), 154-157, 2007.

Appendix

Encoding function over $\mathbb{F}_q$ Let n, q and ω be any fixed positive integers. A constant weight encoding bijective function ϕ_q is described in Algorithm 6.1, this function takes its input from the interval $[0, (q-1)^\omega \binom{n}{\omega}[$ and outputs a q-ary word of length n and Hamming weight ω. Algorithm 6.1 uses a *binary encoder* method introduced by Biswas and Sendrier [14]. This function is a constant weight encoding function taking $s = \omega \log_2(n/\omega)$ input bits and outputting a binary word of length n and weight ω, and which is very efficient because of its linear time encoding.

Algorithm 6.1 q-ary EnumDecoding

Input: integers n, q, ω and $x \in [0, (q-1)^\omega \binom{n}{\omega}[, (\omega \leq n)$
Output: q-ary word of length n and Hamming weight ω

1: $db \leftarrow \lfloor x/(q-1)^\omega \rfloor$
2: ret $\leftarrow$ binary encoder(db, n, ω)
3: $rest \leftarrow x$ **mod** $(q-1)^\omega$
4: **for** i from 1 to n **do**
5: if 0 ¡ ret[i]
6: ret[i] $\leftarrow$ ($rest$ **mod** $(q-1)) + 1$
7: $rest \leftarrow \lfloor rest/(q-1) \rfloor$
8: end if
9: end **for**
10: **Return** ret