Post-Quantum Cryptography: Code-based Signatures

Post-quantum Cryptography: Code-Based Signatures

Pierre-Louis Cayrel and Mohammed Meziani

CASED – Center for Advanced Security Research Darmstadt
Mornewegstrasse, 64293 Darmstadt, Germany
pierre-louis.cayrel@cased.de, mohammed.meziani@cased.de

Abstract. This survey provides a comparative overview of code-based signature schemes with respect to security and performance. Furthermore, we explicitly describe serveral code-based signature schemes with additional properties such as identity-based, threshold ring and blind signatures.

Keywords: post-quantum cryptography, coding-based cryptography, digital signatures.

1 Introduction

Secure digital signature are essential components of IT-security solutions, and several schemes, such as the Digital Signature Algorithm DSA and the Elliptic Curve Digital Signature Algorithm ECDSA are already used in practice. The security of such schemes relies on the hardness of the discrete logarithm problem, either in the multiplicative group of a prime field, or in a subgroup of points of an elliptic curve over a finite field. These computational assumptions, however, could be broken in a quantum setting by Shor's algorithm [39], which was proposed in 1997. Moreover, this algorithm succeeds in polynomial time. Therefore, new, quantum-attack-resistant signature schemes must be designed. Code-based cryptosystems are promising alternatives to classical public key cryptography, and they are believed to be secure against quantum attacks. Their security is based on the conjectured intractability of problems in coding theory, such as the syndrome decoding problem, which has been proven to be NP-complete by Berlekamp, McEliece, and Van Tilborg [4].

In 1978, McEliece [28] first proposed an asymmetric cryptosystem based on the coding theory, which derives its security from the general decoding problem. The general idea is to first select a particular (linear) code for which an efficient decoding algorithm is known, and then to use a trapdoor function to disguise the code as a general linear code. Though numerous computationally-intensive attacks against the scheme appear in the literature [5,20], no efficient attack has been found to date.

The McEliece encryption scheme is not invertible, and therefore it cannot be used for authentication or for signature schemes; this is indeed why very few

T.H. Kim and H. Adeli (Eds.): AST/UCMA/ISA/ACN 2010, LNCS 6059, pp. 82–99, 2010.

signature schemes based on coding theory have been proposed. This problem remained open until 2001, when Courtois et al. [15] showed how to achieve a code-based signature scheme whose security is based on the syndrome decoding problem. While this problem is NP-complete, constructions based on it are still inefficient for large numbers of errors.

A few code-based signature schemes with additional properties, most of them based on the construction of [15], have recently been published. Lattice-based digital signature schemes for a post-quantum age are described in [8]. This paper describes code-based solutions.

Contribution and Organisation
After recalling some basic definitions and notations in Section 2, we discuss the various code-based signature schemes, starting with CFS, Stern, and KKS in Section 3. In Section 4, we describe *all* code-based signature schemes with additional properties, and we conclude in Section 5.

2 Coding Theory Background

This section recalls some basic definitions and then lists some instances of hard problems in coding theory.

Definition 1: ***(Linear Code).*** *An (n, k)-code over $\mathbb{F}_q$ is a linear subspace $\mathcal{C}$ of the linear space $\mathbb{F}_q^n$. Elements of $\mathbb{F}_q^n$ are called* words*, and elements of $\mathcal{C}$ are* codewords*. We call n the* length*, and k the* dimension *of $\mathcal{C}$.*

Definition 2: ***(Hamming distance, weigth).*** *The* Hamming distance $d(\mathsf{x}, \mathsf{y})$ *between two words x, y is the number of positions in which x and y differ. That is, $d(\mathsf{x}, \mathsf{y}) = |\{i \ : \ x_i \neq y_i\}|$, where $\mathsf{x} = (x_1, \ldots, x_n)$ and $\mathsf{y} = (y_1, \ldots, y_n)$. Here, we use $|S|$ to denote the number of elements, or cardinality, of a set S. In particular, $d(\mathsf{x}, \mathbf{0})$ is called the Hamming weigth of x, where $\mathbf{0}$ is the vector containing n 0's. The* minimum distance *of a linear code $\mathcal{C}$ is the minimum Hamming distance between any two distinct codewords.*

Definition 3: ***(Generator matrix).*** *A* generator matrix *of an (n, k)-linear code $\mathcal{C}$ is a $k \times n$ matrix G whose rows form a basis for the vector subspace $\mathcal{C}$. We call a code* systematic *if it can be characterized by a generator matrix $\mathcal{C}$ of the form $\mathsf{G} = (\mathsf{I}_{\mathsf{k}\times\mathsf{k}} | \mathsf{A}_{\mathsf{k}\times(\mathsf{n}-\mathsf{k})})$, where $\mathsf{I}_{\mathsf{k}\times\mathsf{k}}$ is the $k \times k$ identity matrix and A, an $k \times (n-k)$ matrix.*

Definition 4: ***(Parity-check matrix).*** *A* parity-check *matrix of an (n, k)-linear code $\mathcal{C}$ is an $(n-k) \times n$ matrix H whose rows form a basis of the orthogonal complement of the vector subspace $\mathcal{C}$, i.e. it holds that, $\mathcal{C} = \{\mathsf{c} \in \mathbb{F}_q^n \ : \ \mathsf{H}\mathsf{c}^\mathsf{T} = \mathbf{0}\}$.*

In what follows, we recall several NP-complete problems in coding theory. Note that NP-completeness ensures the impossibility to solve a problem in polynomial time *in the worse case*. In other words, the property ensures the existence of *some* hard instances, not the hardness of *every* instance.

Definition 5: ***(Binary Syndrome Decoding (SD) problem)***

- ***Input:*** *An* $r \times n$ *matrix* H *over* $\mathbb{F}_2$*, a target binary vector* $\mathsf{s} \in \mathbb{F}_2^r$*, and an integer* $t > 0$.
- ***Question:*** *Is there a binary word* $\mathsf{x} \in \mathbb{F}_2^n$ *of weight* $\leq t$*, such that* $\mathsf{s} = \mathsf{H}\mathsf{x}^\mathsf{T}$ *?*

This problem has been proved to be NP-Complete by Berlekamp, McEliece, and van Tilborg [4]. In 1994, Barg [2] extended this result of Berlekamp, McEliece, and van Tilborg over $\mathbb{F}_q$ by proving that the following problem, called q-ary Syndrome Decoding (q-SD) problem, is NP-complete.

Definition 6: ***(q-ary Syndrome Decoding (q-SD) problem)***

- ***Input:*** *An* $r \times n$ *matrix* H *over* $\mathbb{F}_q$*, a target vector* $\mathsf{s} \in \mathbb{F}_q^r$*, and an integer* $t > 0$.
- ***Question:*** *Is there a word* $\mathsf{x} \in \mathbb{F}_q^n$ *of weight* $\leq t$*, such that* $\mathsf{s} = \mathsf{H}\mathsf{x}^\mathsf{T}$ *?*

To end this section, we state the Goppa Code Distinguishing (GD) problem which has been proved NP-complete in [19].

Definition 7: ***(Goppa Code Distinguishing (GD) problem)***

- ***Input:*** *An* $(n-k) \times n$ *binary matrix* H.
- ***Question:*** *Is* H *a parity check matrix of a* (n,k)*-Goppa code or of a random* (n,k)*-code ?*

3 Code Based Signature Schemes

During the last twenty years several (linear)-code-based signature schemes were proposed; the first attempts were due to Xinmei Wang [43], followed by Harn and Wang [24] and Alabbadi and Wicker [1]. Unfortunately, the security of these constructions cannot be reduced to the hardness of the problems above, and the schemes were proved insecure [43,24].

Several signature schemes based on these problems were subsequently designed; we outline these below.

3.1 Courtois et al.'s Scheme

Unlike RSA, one of the major obstacles to the widespread use of the McEliece or the Niederreiter cryptosystems was the one-to-one nature of the encryption algorithms, i.e, a random word $x \in \mathbb{F}_2^n$ that is encrypted to, say, y is not necessary decodable. That is, the Hamming distance between y and any codeword is greater than the error capability of the code. This is due the fact that the cardinality of decodable words is very small. To fix this problem, Courtois, Finiasz, and Sendrier [15] (CFS) suggested a method, named *complete decoding*, which increases the correction capability in order to find the nearest word to a given codeword with high probability.

The CFS signature scheme uses Goppa codes that are subfield subcodes of particular alternant code [27]. For given integers m and t, binary Goppa codes are of length $n = 2^m$, of dimension $k = n - mt$, and are t-correcting. The basic idea of the CFS signature scheme is to find parameters n, k, and t such that the Niederreiter scheme described in Algorithm 1 is practically invertible.

Algorithm 1. The Niederreiter PKC

```
Key Generation:
  - Consider an $(n, k)$-code $\mathcal{C}$ over $\mathbb{F}_q$ having a decoding algorithm $\gamma$.
  - Construct an $(n - k) \times n$ parity check matrix $\widetilde{H}$ of $\mathcal{C}$.
  - Choose randomly an $(n - k) \times (n - k)$ invertible matrix $Q$ over $\mathbb{F}_q$.
  - Choose randomly an $n \times n$ permutation matrix $P$ over $\mathbb{F}_q$.
    - The public key: $H = Q\widetilde{H}P$
    - The private key: $(P, \widetilde{H}, Q, \gamma)$
Encryption: To encrypt a message $x \in \mathbb{F}_q^n$ of weight $t$
  - Compute $y = Hx^T$.
Decryption: To decrypt a cipher $y \in \mathbb{F}_q^{n-k}$ s.t. $y = Hx^T$
  - Compute $Q^{-1}y$ $(= \widetilde{H}Px^T)$
  - Find $Px^T$ from $Q^{-1}y$ by applying $\gamma$
  - Find $x$ by applying $P^{-1}$ to $Px^T$.
```

A CFS signature on a message M – see algorithm 2 – is generated by hashing M to a syndrome and then trying to decode it. However, for a t-error correcting Goppa code of length $n = 2^m$, only about $1/t!$ of the syndromes are decodable. Thus, a counter is appended to M, and the signer updates the counter until the hash value is decodable. The signature consists of both the syndrome's weight t error pattern and the counter value.

Algorithm 2. The CFS signature

```
Key Generation:
- Pick random parity check matrix $\widetilde{H}$ of $(n, k)$-binary,
  $t$ error-correcting Goppa code with decoding algorithm $\gamma$.
- Construct binary matrices $Q$, $H$ and $P$ as in Algorithm 1.
Signature: To sign a message $M$
(1) $i \leftarrow i + 1$
(2) $x' = \gamma\left(Q^{-1}h(h(m)\|i)\right)$
(3) if no $x'$ was found go to 1
- Output $(i, x'P)$
Verification:
- Compute $s' = Hx'^T$ and $s = h(h(m)\|i)$.
- The signature is valid if $s$ and $s'$ are equals.
```

Security. The authors of [20] show an attack against the CFS scheme due to Daniel Bleichenbacher. This attack is based on an 'unbalanced' Generalized Birthday Attack. Therefore, the values of m and t used by CFS have been changed. For a security of more than 2^{80} binary operations, [20] proposed new parameters of: $m = 21$ and $t = 10$; $m = 19$ and $t = 11$; or $m = 15$ and $t = 12$. Furthermore, the authors of modified CFS (mCFS) [16] give a security proof in the random oracle model, where the counter is randomly chosen in $\{1, \dots, 2^{n-k}\}$.

3.2 Stern's Identification Scheme

In 1993, Stern [41] presented a 3-pass zero-knowledge protocol which is closely related to the Niederreiter cryptosystem. This protocol aims at enabling a *prover* P to identify himself to a *verifier* V. Its principle is as follows: Let H be an $(n-k) \times n$ binary matrix common to all users, where n and k are integers s.t. $k \leq n$. Each prover P has an n-bit secret key s of weight t and an $(n-k)$-bit *public identifier* y satisfying $\mathsf{y} = H\mathsf{s}^T$. When P needs to authenticate to V as the owner of y, then P and V run the Algorithm 3. It was shown in [41] that the probability that an adversary successfully impersonates an honest prover is 2/3.

Algorithm 3. Stern's Scheme

```
Key Generation : Given binary random $(k, n)$-code with parity-check matrix $H$,
secure hash function $h$.
  - Private key: $s \in \mathbb{F}_2^n$, such that $\mathsf{w}(s) = t$
  - Public key: $y \in \mathbb{F}_2^{n-k}$, such that $Hs^T = y$
Commitments:
  - $P$ chooses randomly $u$ from $\mathbb{F}_2^n$ and $\sigma$ permutation over $\{1, \dots, n\}$
  - $P$ computes the commitments $c_1$, $c_2$, and $c_3$ as follows:
      $c_1 = h(\left(\sigma, Hu^T\right))$, $c_2 = h(\sigma(u))$, $c_3 = h(\sigma\,(u \oplus s))$
  - $P$ sends $c_1$, $c_2$, and $c_3$ to $V$
Challenge: $V$ randomly chooses $b \in \{0, 1, 2\}$ and sends it to $P$
Response:
  - If $b = 0$: $P$ sends $u$ and $\sigma$ to $V$
  - If $b = 1$: $P$ sends $u \oplus s$ and $\sigma$ to $V$
  - If $b = 2$: $P$ sends $\sigma(u)$ and $\sigma(s)$ to $V$
Verification :
  - If $b = 0$: $V$ checks if $c_1$ and $c_2$ were honestly computed
  - If $b = 1$: $V$ checks if $c_1$ and $c_3$ were honestly computed
  - If $b = 2$: $V$ checks if $c_2$ and $c_3$ were honestly computed and $\mathsf{w}(\sigma(s)) = t$
```

In 1995, Véron [42] proposed a dual version of Stern's scheme, which, unlike other schemes based on the SD problem, uses a generator matrix of a random binary linear code. This allows, among other things, for an improved transmission rate.

It is possible to convert Stern's construction into a signature algorithm using the Fiat-Shamir method [18]: the verifier-queries are replaced by values suitably derived from the commitments and the message to be signed. In this case, however, the signature is large, of roughly 120 Kbits.

A variation of the Stern construction using double circulant codes is proposed in [21]. The circulant structure of the public parity-check matrix allows for an easy generation of the whole binary matrix with very little memory storage. They propose a scheme with a public key of 347 bits and a private key of 694 bits. We can also imagine a construction based on quasi-dyadic codes as proposed in [30].

A secure implementation [11] of Stern's scheme uses quasi-circulant codes. This scheme also inherits Stern's natural resistance to leakage attacks such as SPA and DPA.

3.3 Kabatianskii et al.'s Scheme

Kabatianskii, Krouk, and Smeets (KKS) [25] proposed a signature scheme based on arbitrary linear error-correcting codes. Actually, they proposed three versions (using different linear codes) presented in the sequel and all have one point in common: the signature is a codeword of a linear code. We give a full description of the KKS scheme which is illustrated in Algorithm 4.

First consider a code $\mathcal{C}$ defined by a random parity-check matrix H; let d be a good estimate of its minimum distance. Next, consider a linear code $\mathcal{U}$ of length $n' \leq n$ and dimension k defined by a generator matrix $G = [g_{i,j}]$. We suppose that there exist integers t_1 and t_2 s.t. $t_1 \leq \mathsf{w}(u) \leq t_2$ for any non-zero codeword $u \in \mathcal{U}$.

Let J be a subset of $\{1, \ldots, n\}$ of cardinality n', $H(J)$ be the sub matrix of H consisting of the columns h_i where $i \in J$, and define an $r \times n'$ matrix $F \overset{\text{def}}{=} H(J)G^T$. Define a $k \times n$ matrix $G^* = [g^*_{i,j}]$ with $g^*_{i,j} = g_{i,j}$ if $j \in J$ and $g^*_{i,j} = 0$ otherwise. The KKS-signature is $\sigma = mG^*$ for any $m \in \mathbb{F}_q^k$. The main difference with Niederreiter signature occurs in the verification step where the receiver checks that: $t_1 \leq \mathsf{w}(\sigma) \leq t_2$ and $F \cdot m^T = H \cdot \sigma^T$.

Algorithm 4. The KKS Signature

```
Key Generation:
- Pick random $(n, n-r)$ code $\mathcal{C}$, then choose secretly and randomly:
    (1) Generator matrix $G$ of an $(n', k)$ code $\mathcal{U}$ with $n' < n$ and such that $\forall v \in \mathcal{V}, v \neq$
0   $t_1 \leq \mathsf{w}(v) \leq t_2$
    (2) Subset $J$ of $\{1, \cdots, n\}$ of cardinality $n'$
- Form the submatrix $H(J)$ consisting of the columns $h_i$ of a parity check matrix $H$
of $\mathcal{C}$ where $i \in J$
- Define the matrix $F$ as $F = H(J)G^T$.
    Private key: $(J, G)$
    Public Key: $(F, H, t_1, t_2)$
Signature: To sign a message $m$
    (1) Calculate $\sigma^* = m \cdot G$
    (2) Produce $\sigma$ such that
        $\sigma_i = \begin{cases} \sigma_i^* & \text{if} \quad i \in J \\ 0 & \text{if} \quad j \notin J \end{cases}$
Verification: Given $(\sigma, m)$ test whether the following holds:
    (1) $H \cdot \sigma^T = F \cdot m$
    (2) $t_1 \leq \mathsf{w}(\sigma) \leq t_2$
```

Security. The authors of [25] proposed four KKS-signature schemes: KKS-1, KKS-2, KKS-3, KKS-4, which are claimed to be as secure as the Niederreiter scheme if the public parameters do not provide any information. Unfortunately, in [12] the author showed that a generated KKS-signature discloses a lot of information about the secret set J, and so an adversary can find the secret matrix G with a very high probability. Indeed, an attacker needs about 2^{77} binary operations and at most 20 signatures to break the original KKS-3 scheme. For this reason, the authors of [12] suggest new parameters for a security of 40 signatures, as follows: $n = 2000$, $k = 160$, $n' = 1000$, $r = 1100$, $t_1 = 90$ and $t_2 = 110$.

4 Code Based Signature Schemes with Additional Properties

There exist just a few code-based signature schemes with special properties (SP) up to date, namely blind, (threshold) ring signatures, and identity-based signature schemes. By comparison, classical cryptography includes more than sixty classes of signature schemes, some with special properties such as group- or proxy signature. This variety reflects the wide range of application scenarios.

In recent years, existing signature schemes were combined with specific protocols in order to achieve enhanced code-based constructions with additional features, such as anonymity. The properties of the underlying basic scheme could be e.g. authentication and non-repudiation. In what follows, we give a state of the art of such signature schemes.

4.1 Ring Signatures

The concept of ring signatures was firstly introduced in 2001 by Rivest, Rivest, and Tauman [34]. Such signature schemes allow signers of a document to remain anonymous in a group of users, called a *ring*. As opposed to group signatures, no group manager, group setup procedure, cooperation, and revocation mechanisms are needed in ring signatures: the signer specifies an arbitrary ring and then signs on its behalf without permission or assistance from other users. To generate a valid signature, users need their private keys and some other members' public keys.

Zheng et al.'s scheme. In [45], Zheng, Li, and Chen (ZLC) proposed the first code-based ring signature, which extends the CFS signature scheme and is based on the syndrome decoding problem. To describe the ZLC signature, we use the following notations. Let N and l be the number of potential signers and of signers participating in the signature-generating, respectively. Denote by S_i and S_r a potential signer and the ring signer, respectively. Let M be a message and h, a hash function of range $\mathbb{F}_2^{n-k}$. Write the concatenation of s_1 and s_2 as $(s_1|s_2)$; let $u \xleftarrow{R} \mathcal{U}$ indicate that u is randomly selected from a set $\mathcal{U}$. The ring signer and all other potential signers run Algorithm 5 to generate a ring signature on M.

Algorithm 5. The ZLC ring Signature

```
Key Generation: Potential signers $S_i$ generate their private/public keys as in the CFS algorithm (Alg.2):
    - The public key: $H_i = Q_i\widetilde{H_i}P_i$
    - The private key: $(P_i, \widetilde{H_i}, Q_i, \gamma_i)$
Signature: To sign message $M$
    (1) Initialization: For $j = 0, 1, 2, \cdots$
        - $\bar{x}_j \xleftarrow{R} \{0,1\}^{n-k}$
        - Set $x_{r+1,j} = h(N|h(M)|\bar{x}_j)$
    (2) Generating ring sequences: For $j = 0, 1, 2, \cdots$
        - $z_{i,j} \xleftarrow{R} \{0,1\}^n$ s.t. $\mathsf{w}(z_{i,j}) = t$
        - Set $x_{i+1,j} = h\left(N|h(M)|H_i \cdot z_{i,j}^T \oplus x_{i,j}\right)$
    (3) Find an $j_0$ s.t. $x_{r,j_0} \oplus \bar{x}_{j_0}$ is decodable
    (4) Apply the decoding algorithm to get an $z_{r,j_0}$ s.t. $H_r \cdot z_{r,j_0}^T = x_{r,j_0} \oplus \bar{x}_{j_0}$
    (5) Compute the index $I_{z_{i,j_0}}$ corresponding to $z_{i,j_0}$
    (6) The ring signatutre: $(x_{0,j_0}, I_{z_{1,j_0}}, \cdots, I_{z_{l-1,j_0}})$
Verification: Given $(x_{0,j_0}, I_{z_{1,j_0}}, \cdots, I_{z_{(l-1),j_0}})$
    (1) Derive $z_{i,j_0}$ from $I_{z_{i,j_0}}$ for each $i \in \{0, 1, \cdots, l-1\}$
    (2) Compute $x_{i+1,j_0} = h\left(N|h(M)|H_i \cdot z_{i,j_0}^T \oplus x_{i,j_0}\right)$ for $i \in \{0, 1, \cdots, l-1\}$
    (3) Accept if $x_{l,j_0} = x_{0,j_0}$ and reject otherwise.
```

Security and Efficiency. The ZLC scheme is based on CFS signatures, whose security relies on two assumptions: It is hard to solve an instance of the SD problem, and it is hard to distinguish a Goppa code from a random one – the GD problem. The authors of [45] also showed that the ZLC construction provides unforgeability and anonymity. Indeed, the probability of forging a signature is $\frac{1}{2^n}$, and any adversary outside the ring cannot guess the signer's identity due to the uniform distribution of x_{i,j_0}. This scheme is as efficient as the CFS signature, and verification takes tl column operations[1] and $l + 1$ hash computations; the total signature length is close to $(n - k) + \log_2(\binom{n}{t})l$ bits, where $\log_2(\binom{n}{t})$ is the number of bits required to address a word of length n and weight t. For instance, for $m = 16$ and $t = 9$, the signature length is about $144 + 126l$ bits.

4.2 Threshold Ring Signatures

Since its introduction in 2001, a lot of effort has gone into modifying and extending the ring signature scheme [34]. One such extension is the BSS threshold ring signature scheme first proposed by Bresson, Stern and Szydlo [7] in 2002. In threshold ring signature schemes, the secret signing key is distributed amongst N members; at least l of these members are required to generate a valid signature. More precisely, in an (l, N) threshold signature scheme, any set of l members can generate an l-out-of-N signature on behalf of the whole group, without revealing their identity. This type of construction decreases the cost of signing, as it does not require the participation of all N members.

[1] One column operation is one access to a table plus one operation like a comparison or an addition.

Several threshold ring signatures have followed [7]. For example, Wong et al. [44] proposed the tandem construction, a threshold signature scheme using a secure multiparty trapdoor transformation. The threshold ring signature in [26] uses both RSA- and DL-based public keys at the same time and introduces the notion of separability: all signers can select their own keys independently, with distinguishable parameter domains. These signatures, however, and many others, are factoring- ECC-, or pairing-based. Only two coding-based proposals are known, however, up to date. In the following, we outline these proposals.

Aguilar et al.'s scheme (ACG). The first non-generic code-based threshold ring signature scheme is introduced in [29]; it generalizes Stern's identification protocol into a threshold ring signature scheme, using the Fiat-Shamir paradigm [18]. Algorithm 6 explains how Aguilar et al.'s construction works. We denote by N the number of signers (provers) in the ring, and let l with $(l \leq N)$ stand for the number of first signers. A leader $\mathrm{S_L}$ amongst them gives to the ring members their public keys.

Security and Efficiency. Aguilar et al.'s identification scheme is a zero-knowledge protocol with a cheating probability of 2/3 as in Stern's scheme. Its security relies on the hardness of the SD problem: finding a vector $s \in \mathbb{F}_2^{nN}$ of weight tl and a null syndrome w.r.t. H such that each block (out of N) of length is of weight t or 0. The signing complexity and signature length are N times those of Stern's signature scheme: a complexity of about $140n^2N$ independently of l, and a length of about $20kB \times N$. In order to reduce the public key size, [29] suggested the use of double-circulant matrices, requiring $nN/2$, rather than $n^2N/2$ storage bits. For double-circulant matrices, [21] proposes parameters $n = 347$ and $t = 76$ for an 83-bit security level, rather than $n = 634$, $t = 69$, and a rate 1/2, as in an 80-bit secure Stern's scheme.

Dallot et al.'s scheme . A second code-based threshold ring signature has been proposed by Dallot and Vergnaud (DV) in [17], combining the generic construction of Bresson et al. [7] with the CFS signature scheme. The DV construction requires the following: an (n, k) t-error-correcting binary Goppa code with $n = 2^m$ and $k = n - mt$, where m a positive integer. We denote by N and l the number of ring users and the number of signers respectively. Let h be a public collision-resistant hash function of range $\{0,1\}^{mt}$, $f_{(\cdot)}$, a trapdoor one-way function : $\{0,1\}^a \rightarrow \{0,1\}^{mt}$, and $(E_{k,i})$, a family of random permutations that encrypts b-bit messages with a_0-bit keys and an additional parameter $i \in [1, N]$. We again denote concatenation as $(s|s')$ and random selection by $x \overset{R}{\longleftarrow} \mathcal{S}$.

For simplicity, we index the signers as $1, \cdots, l$. In addition, each ring member i is associated with a secret/public key pair as in the CFS construction, i.e. the public key is $H_i = Q_i\widetilde{H_i}P_i$ and the secret key is $(P_i, \widetilde{H_i}, Q_i, \gamma_i)$. Dallot et al.'s procedure is presented in Algorithm 7.

Algorithm 6. The ACG Identification Scheme

Key Generation: Each potential signer S_i has:

- **The public key:** the $(n-k) \times n$ binary matrix H_i
- **The private key:** n-bit word s_i the weight t s.t. $H_i s_i^T = 0$,
- **The ring public key :** $(n-k)N \times nN$ binary matrix H defined by:

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

Commitment:

- Each prover S_i (among l signers) chooses randomly $z_i \in \mathbb{F}_2^n$ and a permutation σ_i of $\{1, \cdots, n\}$

- Each prover S_i sends to S_L three commitments $c_{1,i}$, $c_{2,i}$ and $c_{3,i}$ given by:
$c_{1,i} = h(\left(\sigma_i, H_i z_i^T\right))$, $c_{2,i} = h(\sigma_i(z_i))$ and $c_{3,i} = h(\sigma_i\,(z_i \oplus s_i))$

- S_L generates the $N-l$ missing commitments for the $N-l$ non-signers by fixing all remaining s_i at 0.

- S_L chooses randomly a constant n-block permutation Π on N blocks

- S_L computes the master commitments C_1, C_2 and C_3 using $c_{1,i}$, $c_{2,i}$ and $c_{3,i}$ by:

$C_1 = h(\Pi\,(c_{1,1}, \cdots, c_{1,N}))$, $C_2 = h(\Pi\,(c_{2,1}, \cdots, c_{2,N}))$, $C_3 = h(\Pi\,(c_{3,1}, \cdots, c_{3,N}))$

- S_L sends C_1, C_2 and C_3 to the verifier V.

Challenge:

-V sends a challenge $b \in \{0, 1, 2\}$ to S_L which forwards this challenge to l signers.

Response:

- Perform the challenge step of the Stern's protocol between each prover S_i and S_L

- S_L simulates the missing $N-l$ Stern's protocol with $s_i = 0$ for all $l+1 \leq i \leq N$

- S_L gathers all answers to create the global response for V as follows:

  * If $b = 0$: S_L sets $z = (z_1 \cdots, z_N)$, $\Omega = \Pi \circ (\sigma_1, \cdots, \sigma_N)$ and reveals z and Ω
  * If $b = 1$: S_L constructs $x = (y_1 \oplus s_1, \cdots, y_N \oplus s_N)$ and reveals x and Ω
  * If $b = 2$: S_L constructs $\Pi(y_1, \cdots, y_N)$ and reveals $\Omega(s_1, \cdots, s_N)$

Verification:

- If $b = 0$: V checks that $\Omega(s)$ is a n-block permutation and that C_1, C_2 were honestly computed.

- If $b = 1$: V checks that $\Omega(s)$ is a n-block permutation and that C_1, C_3 were honestly computed.

- If $b = 2$: V checks that:

  * C_2, C_3 were honestly computed
  * $\mathsf{w}(\Omega(s)) = lt$
  * each of block of $\Omega(s)$ of length n has weigth t or 0.

Algorithm 7. The DV threshold ring signature scheme

```
Key Generation : Each signer in the ring has to:
  - choose an $(n,k)$-code $\mathcal{C}_i$ over $\mathbb{F}_2$ having a decoding algorithm $\gamma_i$ correcting up to $t$ errors.
  - construct an $n \times (n-k)$ parity check matrix $\widetilde{H_i}$ of $\mathcal{C}_i$.
  - choose randomly an $(n-k) \times (n-k)$ invertible matrix $Q_i$ over $\mathbb{F}_2$.
  - choose randomly an $n \times n$ permutation matrix $P_i$ over $\mathbb{F}_2$.
    - The public key: $H_i = Q_i \widetilde{H_i} P_i$
    - The private key: $(P_i, \widetilde{H_i}, Q_i, \gamma_i)$
Signature: To generate a signature on a message $M$:
  - compute the symmetric key for $E$: $k = h(M)$.
  - compute value at origin: $v_0 = h(H_i, \cdots, H_N)$ .
  - choose random seeds: For each $i = l+1, \cdots, N$ do
      (1) $x_i \xleftarrow{R} \{x \in \mathbb{F}_2^n \quad \text{s.t.} \quad \mathsf{w}(x) \leq t\}$
      (2) $r_i \xleftarrow{R} \{1 \cdots, 2^{tm}\}$
      (3) $y_i \leftarrow H_i x_i^T + h(M|r_i)$
  - compute a sharing polynomial: Find a polynomial $f$ over $\mathbb{F}_{2^{tm}}$ s.t.
      - $\deg(f) = N - l$
      - $f(0) = v_0$
      - $f(i) = E_{k,i}(y_i) \;\; \forall\, l+1 \leq i \leq N$
  - For each $i = 1, \cdots, l$ do
      - $x_i \leftarrow \emptyset$
      - While $x_i = \emptyset$ do
          (1) $r_i \xleftarrow{R} \{1 \cdots, 2^{tm}\}$
          (2) $z_i \leftarrow \gamma_i(Q_i^{-1} \cdot (E_{k,i}(f(i)) + h(M|r_i)))$
          (3) if $z_i \neq \emptyset$ then $x_i \leftarrow z_i P_i^{-1}$
  - The signature: $\sigma = (N, x_1, \ldots, x_N, r_1, \ldots, r_n, f)$
Verification: Given $(N, x_1, \ldots, x_n, r_1, \ldots, r_n, f)$ any user can verify the signature by:
  - Recovering the symmetric key: $k = h(M)$
  - Recovering $(y_i)$: $y_i = H_i x_i^T + h(M|r_i)$
  - checking the equations:
      (1) $f(0) \overset{?}{=} h(H_1, \cdots, H_N)$
      (2) $f(i) \overset{?}{=} E_{k,i}(y_i) \;\; \forall\, 1 \leq i \leq N$
```

Security and Efficiency. The DV construction is a provably secure threshold ring signature satisfying three properties: consistency, anonymity, and unforgeability [17]. Unforgeability is proved based on two coding theory problems. One is the well known NP-complete [4] Bounded Distance Decoding problem (GBDP) which is a variant of the SD problem with the constraint that the number of errors is up to $(n - k/\log_2(n))$ as in the mCFS signature scheme. The second is the GCD problem: distinguishing a randomly sampled Goppa code from a random linear code (with the same parameters); this problem is widely considered as difficult [36]. The complete security proof of the DV scheme is in [17].

For a ring with N members, the set of public keys (H_i) are stored in $n(n-k)N$ bits. To produce a valid signature, the signer has to perform the following calculations: computing $N-l$ syndromes, N polynomial evaluations that can be performed in $2N(N-l)$ binary operations using Horner's rule and $l(t!)$ decodings of Goppa codes, each consisting of: computing a syndrome (in about $t^2m^2/2$ binary operations), computing a localisator polynomial ($6t^2m$ binary operations) and computing its roots ($2t^2m^2$ binary operations). Thus, the total cost for generating a signature would be $(N-l)t^2m^2/2 + 2N(N-l) + l(t!)(3/2 + 6/m)$ binary operations.

Signature verification requires $(N + 1)$ polynomial evaluations N syndrome-computations, resulting in $2(N + 1)(N - l) + Nt^2m^2/2$ binary operations. The signature consists of: the number N of ring-users, which are stored in $\log_2(N)$ bits, N random vectors x_i of weight up to t which can be indexed with a $\lfloor \log_2 \sum_{i=1}^{t} \binom{2^m}{i} \rfloor$ bit counter, N random vectors r_i in $\{0, \ldots, 2^{mt} - 1\}$ requiring at most mt bits and a polynomial of degree $N - l$ which needs $(N - l + 1)mt$ bits. The signature size is thus about $N \left(\lfloor \log_2 \sum_{i=1}^{t} \binom{2^m}{i} \rfloor + 2mt \right) + \log_2(N) - (l-1)mt$ bits.

4.3 Blind Signatures

Blind signatures were first introduced by Chaum [13] for applications such as e-Voting or electronic payment systems, which require anonymity. The main goal of Chaum's scheme is to ensure *Blindness* (i.e., the signed message is disguised – blinded – before signing) and *Untraceability* (i.e., the signer cannot trace the signed message after the sender has revealed the signature publicly).

Several blind signature schemes followed Chaum's proposal. In 1988, the authors of [14] showed a new signature scheme for electronic payment systems. Later, the authors of [40] introduced fair blind signature schemes. In 1992, another blind signature scheme based on factoring and discrete logarithm-based identification schemes [31] have been developed. Based on Schnorr's [35] and Guillou-Quisquater's [23] protocols, provably secure blind signature schemes were presented in [33]. As far as we know, there exists only a single code-based blind signature scheme, namely Overbeck's construction [32].

Overbeck's scheme. The general idea behind Overbeck's protocol is, instead of blinding the message, to use permuting kernels in order to blind the signer's public key from a public key of a code. A blind signature is thus generated by the owner of a valid secret key, with the blinded public key. During verification, the blinder gives a static zero-knowledge proof showing that the private and public keys are paired. This proof is based on the Permuted Kernels Problem (PKP) which can be formulated as follows: Given a random (n, k) code and a random permuted subcode of dimension $L < k$, find the permutation. This problem is known to be NP-hard in the general case [38].

For simplicity, we denote a code by its generator matrix. Let h be a hash function, r be a random seed, and w be a positive integer. Denote by PKP-proof(A, B) the static PKP-Proof that code A is an isometric subcode of code B, s.t. $dim(A) \leq dim(B)$. The notation $dim(C)$ stands for the dimension of the code C.

A slightly modified version of Overbeck's blind signature scheme is depicted in Algorithm 8.

Algorithm 8. Overbeck's blind signature

```
Key Generation:
  - choose an $(n,k)$-code $\mathcal{C}$ over $\mathbb{F}_2$ having a decoding algorithm $\gamma$ correcting up to $t$ errors.
  - construct an $(n-k)\times n$ parity check matrix $\widetilde{H}$ of $\mathcal{C}$.
  - choose randomly an $(n-k)\times(n-k)$ invertible matrix $Q$ over $\mathbb{F}_q$.
  - choose randomly an $n\times n$ permutation matrix $P$ over $\mathbb{F}_q$.
    - The public key: $H = Q\widetilde{H}P$
    - The private key: $(P,\widetilde{H},Q,\gamma)$
Blinding: The user has to:
  - generate a random $p\times n$ public matrix $R_0$ over $\mathbb{F}_q$
  - generate a random $L\times p$ matrix $K$ of full rank over $\mathbb{F}_q$
  - set $R = KR_0$
  - generate a $n\times n$ permutation matrix $\Pi$
  - create the blind generator matrix $G_b$ as follows: $G_b = [\frac{G}{R}]\Pi$
  - derive from $G_b$ the blind check matrix $H_b$
  - solve $H_b x^T = h(M|H_b)$ in $x$
  - output $s = H(x\Pi^{-1})$ (the blind syndrome) and $u = (r,\Pi,H_b)$ (the unblinding information)
Unblinding: Given $M$, $s$ and a correct signature $\sigma$ of $H$:
  - Check if that: $\mathsf{w}(\sigma) = t$ and $H\sigma^T = s$. If not output failure.
  - Generate a PKP-Proof($H_b,H_0$) with $H_0 = [\frac{H}{R_0}]$
  - Output the blind signature $\sigma_b = (r,H_b,\sigma\pi,$PKP-Proof$(H_b,H_0))$
Verification: Given $r$, $M$, and $\sigma_b = (r,H_b,\sigma\pi,$PKP-Proof$(H_b,H_R))$, where $H_R$ is a parity-check matrix of the code generated by $R$, verify $\sigma_b$ by:
  - Generate the matrix $R_0$ from $r$
  - Find some vector $\tau$ satisfying $H\tau^T = h(M|H_b)$
  - Verify $\mathsf{w}(\sigma\Pi) < t$ and $(\tau - \sigma\Pi) \in H_b$
  - Check PKP-Proof($H_b,H_R$)
```

Security and Efficiency. Overbeck assessed the efficiency of his scheme by applying it to the CFS construction. For a $(2^m, 2^m - mt)$ binary Goppa code, the complexity of this scheme is as follows: To store a public parity check matrix key H $2^m \times mt$ bits are needed. The blind matrix H_b is a $2^m \times (mt - L)$ binary matrix. To generate a single signature, the Blinding algorithm is run $\left(2^{mt}/\binom{2^m}{t}\right)$ times, each time requiring m^3t^2 binary operations for the signer and m^3t^3 for the blinder. Thus, the total signing complexity is about $\left(2^{mt}/\binom{2^m}{t}\right)(m^3t^2 + m^3t^3)$ binary operations per signature. The blind signature size mainly depends on PKP-Proof(H_b, H_0) requiring the storage of the generator matrix G_b of size $((k+L)\times n)$ bits in each round.

The author of [32] does not explicitly prove the proposed construction, but he claims that the scheme is provably secure based on the hardness of some instances of the PKP and SD problems.

4.4 Identity-Based Signatures

Identity-based cryptography was proposed by Shamir in 1984 [37] so as to simplify PKI requirements. An identity is associated with data such as an e-mail or IP-address instead of a public key; the secret key is issued by a trusted Key Generation Center (KGC) thanks to a master secret that only the KGC knows. Some PKI and certificate costs can now be avoided. However, identity-based cryptography suffers from a major drawback: the KGC must be trusted completely. A solution to this problem, also known as the key escrow problem, is to employ multiple PKGs that jointly produce the master secret key (see [6]).

Identity-based cryptography has led to the development, in 1984, of identity-based signature (IBS) schemes. One of the most interesting contributions to this subject is the framework of [3], for which a large family of IBS are proved secure. This work was later extended in [22], which implied the existence of generic IBS constructions with various additional properties that are provably secure in the standard model.

Cayrel et al.'s identification scheme. The first coding-based IBS appeared in [10] is due to Cayrel, Gaborit and Girault (CGG). The main idea of this scheme is to combine the mCFS scheme with a slightly modified Stern scheme to obtain an IBS scheme whose security relies on the syndrome decoding problem. The mCFS scheme is used to solve an instance of the SD problem given a hash value of an identity in the first step, while Stern's protocol is used for identification in the following step.

Consider a linear (n, k)-code over $\mathbb{F}_2$ with a disguised parity-check matrix H defined by $H = Q\widetilde{H}S$ with $\widetilde{H}$ the original parity-check matrix, Q invertible, and S a permutation matrix. The matrix H is public, while Q and S are kept secret by a trusted Key Generation Center (KGC). Denote by h a hash function with outputs in $\{0,1\}^{n-k}$. In addition, let y be the identity associated to the prover wishing to authenticate to a verifier. The Cayrel et al. identification scheme works as Stern's protocol, with a few variations.

Security and Efficiency. Cayrel et al.'s identification scheme (CFS-Stern IBS) is provably secure against passive (i.e., eavesdropping only) impersonation attacks [10], based on the hardness of the SD and GD problems. The security and the performance of the proposed identification scheme mainly depends on the difficulty of finding a couple $\{s, j\}$ without the description H. At the same time, an attacker needs to minimize the number of attempts used to find j, so as to be able to find s with minimal cost.

4.5 Summary

In Table 1 we summarize the complexity of the code-based proposals with special properties by using the following notations: t is the correction capability of the code, n denotes the code length which equals 2^m in the case of Goppa codes, k indicates the code dimension, N is number of users in the ring, l is number of involved signers

Algorithm 9. The CGG Identity-based scheme

```
Key Deliverance:
  - The prover sends its identity $y$ to KGC
  - TCA runs the CFS algorithm (Alg. 2) on $y$ to get $\{s, j\}$ s.t. $h(h(y)|j) = Hs^T$ with $\mathsf{w}(s) \leq t$
    - The Public key: $h(h(y)|j)$
    - The Private key: $\{s, j\}$
Identification: Run the Stern's protocol as follows:
  - Commitments:
      - $P$ chooses randomly $u$ from $\mathbb{F}_2^n$ and $\sigma$ permutation over $\{1, \ldots, n\}$
      - $P$ computes the commitments $c_1$, $c_2$ and $c_3$ as follows:
            $c_1 = h\left(\sigma, Hu^T\right)$, $c_2 = h(\sigma(u))$, $c_3 = h(\sigma\left(u \oplus s\right))$
      - $P$ sends $c_1$, $c_2$ and $c_3$ and $j$ to $V$
  - Challenge: $V$ choose randomdy $b \in \{0, 1, 2\}$ and sends it to $P$
  - Response:
      - If $b = 0$: $P$ sends $u$ and $\sigma$ to $V$
      - If $b = 1$: $P$ sends $u \oplus s$ and $\sigma$ to $V$
      - If $b = 2$: $P$ sends $\sigma(u)$ and $\sigma(s)$ to $V$
  - Verification :
      - If $b = 0$: $V$ checks if $c_1$ and $c_2$ were honestly computed
      - If $b = 1$: $V$ checks if $c_1$ and $c_3$ were honestly computed
      - If $b = 2$: $V$ checks if $c_3$ and $c_3$ were honestly computed and $\mathsf{w}(s) = t$
```

in the ring , L is dimension of the subcode introduced in [32] and r_i is number of rounds for $i = 1, 2$.

Based on these notations, we define the two following quantities:

– $A(m, t, N, l) = N\left(\lfloor \log_2 \sum_{i=1}^{t} \binom{2^m}{i} \rfloor + 2mt\right) + \log_2(N) - (l-1)mt$
– $B(m, t, N, l) = (N - l)t^2m^2/2 + 2N(N - l) + l(t!)(3/2 + 6/m)$.

For example, for $m = 22, t = 9$,

Table 1. Code-based signatures with special properties with $(m, t, N, l, L, r_1, r_2) = (15, 12, 100, 50, 40, 58, 80)$

Schemes	Pk size in bits	Sign. size in bits	Sign. cost in bops
Identity Based Signatures			
PGGG [9]	$2^m tm$ (≈ 0.7 MB)	$2^m \times r_1$ (≈ 1.1 MB)	$t!t^2m^2(1/2 + 2 + 6/m)(\approx 2^{45})$
Ring Signatures			
ZLC [45]	$2^m tm$ (≈ 0.7 MB)	$tm + \log_2(\binom{2^m}{t})l$ (≈ 0.95 kB)	$t!t^2m^2$ ($\approx 2^{43.8}$)
Threshold(ring) Signatures			
ACG [29]	$n^2N/2$ (≈ 2.41 MB)	$20000 \times N$ (≈ 0.24 MB)	$140n^2N$ ($\approx 2^{32.3}$)
DV [17]	$2^m tmN$ (≈ 70 MB)	$A(m, t, N, l)$ (≈ 5.2 kB)	$B(m, t, N, l)$ ($\approx 2^{35.4}$)
Blind Signatures			
Overbeck[32]	$2^m tm$ (≈ 0.7 MB)	$((2^m - tm + L)2^m) \times r_2$ (≈ 9.95 GB)	$\left(2^{mt}/\binom{2^m}{t}\right)(m^3t^2 + m^3t^3)$ ($\approx 2^{190}$)

5 Conclusion

Several code-based signature schemes already exist, exhibiting features such as small public key size (Stern [41]), short signature size (CFS [15]), or a good balance of public key and signature size at the expense of security (KKS [25]). By combining such schemes, additional constructions such as identity-based, threshold ring, or blind signatures can be obtained. However these schemes also inherit the disadvantages of the underlying protocols. We strongly encourage the code-based research community to actively investigate future possibilities for post-quantum signature schemes, such as multi-signatures, group signatures, or linkable signatures.

References

1. Alabbadi, M., Wicker, S.B.: Digital signature scheme based on error–correcting codes. In: Proc. of 1993 IEEE International Symposium on Information Theory, pp. 19–29. Press (1993)
2. Barg, S.: Some New NP-Complete Coding Problems. Probl. Peredachi Inf. 30, 23–28 (1994)
3. Bellare, M., Chanathip, N., Gregory, N.: Security Proofs for Identity-Based Identification and Signature Schemes. J. Cryptol. 22(1), 1–61 (2008)
4. 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)
5. Bernstein, D.J., Lange, T., Peters, C.: Attacking and defending the McEliece cryptosystem. Cryptology ePrint Archive, Report 2008/318 (2008), `http://eprint.iacr.org/`
6. Boneh, D., Franklin, M.: Identity-based encryption from the Weil pairing, pp. 213–229. Springer, Heidelberg (2001)
7. Bresson, E., Stern, J., Szydlo, M.: Threshold Ring Signatures and Applications to Ad-hoc Groups. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 465–480. Springer, Heidelberg (2002)
8. Buchmann, J., Lindner, R., Ruckert, M., Schneider, M.: Post-Quantum Cryptography: Lattice Signatures (2009)
9. Cayrel, P.-L., Gaborit, P., Galindo, D., Girault, M.: Improved identity-based identification using correcting codes. CoRR, abs/0903.0069 (2009)
10. Cayrel, P.-L., Gaborit, P., Girault, M.: Identity-based identification and signature schemes using correcting codes. In: Augot, D., Sendrier, N., Tillich, J.-P. (eds.) WCC 2007, pp. 69–78 (2007)
11. Cayrel, P.-L., Gaborit, P., Prouff, E.: Secure Implementation of the Stern Authentication and Signature Schemes for Low-Resource Devices. In: Grimaud, G., Standaert, F.-X. (eds.) CARDIS 2008. LNCS, vol. 5189, pp. 191–205. Springer, Heidelberg (2008)
12. Cayrel, P.L., Otmani, A., Vergnaud, D.: On Kabatianskii-Krouk-Smeets Signatures. In: Carlet, C., Sunar, B. (eds.) WAIFI 2007. LNCS, vol. 4547, pp. 237–251. Springer, Heidelberg (2007)
13. Chaum, D.: Blind Signatures for Untraceable Payments. In: CRYPTO, pp. 199–203 (1982)
14. Chaum, D., Fiat, A., Naor, M.: Untraceable Electronic Cash. In: Goldwasser, S. (ed.) CRYPTO 1988. LNCS, vol. 403, pp. 319–327. Springer, Heidelberg (1990)
15. Courtois, N., Finiasz, M., Sendrier, N.: How to Achieve a McEliece-based Digital Signature Scheme. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 157–174. Springer, Heidelberg (2001)

16. Dallot, L.: Towards a Concrete Security Proof of Courtois, Finiasz and Sendrier Signature Scheme (2007), `http://users.info.unicaen.fr/~ldallot/download/articles/CFSProof-dallot.pdf`
17. Dallot, L., Vergnaud, D.: Provably Secure Code-Based Threshold Ring Signatures. In: Cryptography and Coding 2009: Proc. of the 12th IMA International Conference on Cryptography and Coding, pp. 222–235. Springer, Heidelberg (2009)
18. Fiat, A., Shamir, A.: How to prove yourself: practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987)
19. Finiasz, M.: Nouvelles constructions utilisant des codes correcteurs dérreurs en cryptographie à clé publique. PhD thesis, INRIA - Ecole Polytechnique (2004)
20. Finiasz, M., Sendrier, N.: Security Bounds for the Design of Code-based Cryptosystems. To appear in Advances in Cryptology – Asiacrypt 2009 (2009), `http://eprint.iacr.org/2009/414.pdf`
21. Gaborit, P., Girault, M.: Lightweight code-based authentication and signature. In: IEEE International Symposium on Information Theory – ISIT 2007, Nice, France, pp. 191–195. IEEE, Los Alamitos (2007)
22. Galindo, D., Herranz, J., Kiltz, E.: On the Generic Construction of Identity-Based Signatures with Additional Properties. In: Lai, X., Chen, K. (eds.) ASIACRYPT 2006. LNCS, vol. 4284, pp. 178–193. Springer, Heidelberg (2006)
23. Guillou, L.C., Quisquater, J.-J.: A practical zero-knowledge protocol fitted to security microprocessor minimizing both transmission and memory. In: Günther, C.G. (ed.) EUROCRYPT 1988. LNCS, vol. 330, pp. 123–128. Springer, Heidelberg (1988)
24. Harn, L., Wang, D.C.: Cryptoanalysis and modification of digital signature scheme based on error–correcting codes. Electronics Letters 28(2), 157–159 (1992)
25. Kabatianskii, G., Krouk, E., Smeets, B.J.M.: A digital signature scheme based on random error-correcting codes. In: Darnell, M.J. (ed.) Cryptography and Coding 1997. LNCS, vol. 1355, pp. 161–167. Springer, Heidelberg (1997)
26. Liu, J.K., Wei, V.K., Wong, D.S.: A Separable Threshold Ring Signature Scheme. In: Lim, J.-I., Lee, D.-H. (eds.) ICISC 2003. LNCS, vol. 2971. Springer, Heidelberg (2004)
27. MacWilliams, F.J., Sloane, N.J.A.: The theory of error-correcting codes, vol. 16. North-Holland Mathematical Library, Amsterdam (1977)
28. McEliece, R.J.: A public-key cryptosystem based on algebraic coding theory. Jpl dsn progress report 42-44, pp. 114–116 (1978)
29. Aguilar Melchor, C., Cayrel, P.-L., Gaborit, P.: A New Efficient Threshold Ring Signature Scheme Based on Coding Theory. In: Buchmann, J., Ding, J. (eds.) PQCrypto 2008. LNCS, vol. 5299, pp. 1–16. Springer, Heidelberg (2008)
30. Misoczki, R., Barreto, P.S.L.M.: Compact McEliece Keys from Goppa Codes. Preprint (2009), `http://eprint.iacr.org/2009/187.pdf`
31. Okamoto, T.: Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes. In: Brickell, E.F. (ed.) CRYPTO 1992. LNCS, vol. 740, pp. 31–53. Springer, Heidelberg (1993)
32. Overbeck, R.: A Step Towards QC Blind Signatures. Cryptology ePrint Archive, Report 2009/102 (2009), `http://eprint.iacr.org/`
33. Pointcheval, D., Stern, J.: Provably Secure Blind Signature Schemes. In: Kim, K.-c., Matsumoto, T. (eds.) ASIACRYPT 1996. LNCS, vol. 1163, pp. 252–265. Springer, Heidelberg (1996)
34. Rivest, R.L., Shamir, A., Tauman, Y.: How to leak a secret. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, p. 552. Springer, Heidelberg (2001)

35. Schnorr, C.-P.: Efficient Signature Generation by Smart Cards. J. Cryptology 4(3), 161–174 (1991)
36. Sendrier, N.: Cryptosystèmes à clé publique basés sur les codes correcteurs d'erreurs. Mémoire d'habilitation à diriger des recherches, Université Paris 6 (March 2002)
37. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (1985)
38. Shamir, A.: An efficient identification scheme based on permuted kernels. In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 606–609. Springer, Heidelberg (1990)
39. Shor, P.W.: Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. SIAM J. Sci. Statist. Comput. 26, 1484 (1997)
40. Stadler, M., Piveteau, J.-M., Camenisch, J.: Fair Blind Signatures. In: Guillou, L.C., Quisquater, J.-J. (eds.) EUROCRYPT 1995. LNCS, vol. 921, pp. 209–219. Springer, Heidelberg (1995)
41. Stern, J.: A new identification scheme based on syndrome decoding. In: Stinson, D.R. (ed.) CRYPTO 1993. LNCS, vol. 773, pp. 13–21. Springer, Heidelberg (1994)
42. Véron, P.: Improved Identification Schemes Based on Error-Correcting Codes. Appl. Algebra Eng. Commun. Comput. 8(1), 57–69 (1996)
43. Wang, X.M.: Digital signature scheme based on error-correcting codes. Electronics Letters (13), 898–899 (1990)
44. Wong, D.S., Fung, K., Liu, J.K., Wei, V.K.: On the RS-Code Construction of Ring Signature Schemes and a Threshold Setting of RST. In: Qing, S., Gollmann, D., Zhou, J. (eds.) ICICS 2003. LNCS, vol. 2836, pp. 34–46. Springer, Heidelberg (2003)
45. Zheng, D., Li, X., Chen, K.: Code-based Ring Signature Scheme. I. J. Network Security 5(2), 154–157 (2007)