Security Analysis and Efficient Implementation of Code-based Signature Schemes

Partha Sarathi Roy[1], Kirill Morozov[2], Kazuhide Fukushima[1], Shinsaku Kiyomoto[1] and Tsuyoshi Takagi[3]

[1]*Information Security Laboratory, KDDI Research, Inc., 2-1-15 Ohara, Fujimino-shi, Saitama 356-8502, Japan*

[2]*Department of Computer Science and Engineering, University of North Texas, Denton, TX 76207, U.S.A.*

[3]*Department of Mathematical Informatics, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan*

Keywords: Post-quantum Cryptography, Code-based Cryptography, Signature Scheme, Fiat-Shamir Transformation, Identification Scheme.

Abstract: In this paper, we derive code-based signature schemes using Fiat-Shamir transformation on code-based zero-knowledge identification schemes, namely the Stern scheme, the Jain-Krenn-Pietrzak-Tentes scheme, and the Cayrel-Veron-El Yousfi scheme. We analyze the security of these code-based signature schemes and derive the security parameters to achieve the 128-bit level security. Furthermore, we implement these signature schemes and compare their performance on a PC.

1 INTRODUCTION

Digital signatures are essential components of IT-security solutions. Security of the schemes, already used in practice, relies on the number-theoretic hardness assumptions. Unfortunately, these problems are known to be solvable in polynomial time on quantum computers using Shor's algorithm (Shor, 1994). Hence, quantum computers would be able to break popular cryptosystems such as RSA or ElGamal (including its elliptic-curve variant) in polynomial time. Given these circumstances, it is important to consider the transition to post-quantum digital signature schemes. In this work, we focus on the practical use of code-based signature scheme as one of the post-quantum candidates.

CFS (Courtois et al., 2001), KKS (Kabatianskii et al., 1997), and their variants are the most celebrated code-based signature schemes. Unfortunately, there are various drawbacks in respect of security, computation time, key or signature size. The CFS scheme was proven existentially unforgeable under chosen message attack (EUF-CMA) by Dallot (Dallot, 2008) under the hardness of the Goppa-Parametrized Bounded Decoding problem and the Goppa-Code Distinguishing (GD) problem. Even with the existence of a distinguisher (Faugére et al., 2011) for the Goppa codes of high rate, a simple modification can provide Strong EUF-CMA security for the CFS signature (Morozov et al., 2018). However, from the practical prespective, signing time of CFS signature is somewhat high due to the difficulty of finding decodable syndromes. Debris-Alazard et al. (Debris-Alazard et al., 2017) exposed a problem with the recently introduced SURF signature scheme. The main drawback of the KKS-like signature schemes is their security as shown by Otmani et al. (Otmani and Tillich, 2011). Therefore, we will explore the time-tested approach to signature schemes via the Fiat-Shamir transform over zero-knowledge identification schemes.

Our Contribution. In this paper, we study code-based signature schemes via Fiat-Shamir transformation on zero-knowledge (ZK) identification schemes. The Fiat-Shamir transform seems to be the promising avenue to have both efficient and secure code-based signature schemes. We chose to study and compare zero-knowledge identification schemes by Stern (Stern, 1994), Jain et al. (Jain et al., 2012) and Cayrel et al. (Cayrel et al., 2010). Note that Jain et al. (Jain et al., 2012) pointed out a flaw in the proof of zero-knowledge property of Veron's code-based identification scheme (Véron, 1997), and so Jain et al. provided an alternative scheme which is indeed ZK. As security assumption, Jain et al. used the so-called Exact-LPN (xLPN) problem (Jain et al.,

Roy, P., Morozov, K., Fukushima, K., Kiyomoto, S. and Takagi, T.
Security Analysis and Efficient Implementation of Code-based Signature Schemes.
DOI: 10.5220/0007259102130220
In *Proceedings of the 5th International Conference on Information Systems Security and Privacy (ICISSP 2019)*, pages 213-220
ISBN: 978-989-758-359-9

2012), which is, in fact, identical to the general decoding problem considered both in Veron's paper, and in this work.[1] There are some existing studies on the efficient implementation of Stern's scheme by Gaborit et al. (Gaborit and Girault, 2007), and Cayrel et al. (Cayrel et al., 2008). El Yousfi et al. (Alaoui et al., 2013) have studied the implementations of Stern (Stern, 1994), Veron (Véron, 1997) and Cayrel et al. (Cayrel et al., 2010). However, these works show system parameters for identification schemes that achieve 80-bit security. Due to the growing power of modern computing systems, it seems preferable to investigate the performance of the signature schemes for 128-bit security benchmark. We implement in C the digital signatures based on Stern's (Stern, 1994), Jain et al.'s (Jain et al., 2012) and Cayrel et al.'s (Cayrel et al., 2010) identification schemes, and compare their performance on a PC.

2 PRELIMINARIES

2.1 Notations

We will use the following notations throughout the article.

- $\{0,1\}^*$: bit string of arbitrary length.
- $\mathbb{F}_q$: Galois field of q elements.
- $\mathbb{F}_q^n$: Vector with n elements over $\mathbb{F}_q$.
- $\mathbb{F}_q^{m\times n}$: Matrix with m rows and n columns whose elements are in $\mathbb{F}_q$.
- $\mathsf{wt}(c)$: Hamming weight of string c, i.e., the number of non-zero positions of the string.
- $\|$: Concatenation of strings. We see a string as a column vector.

2.2 Zero-knowledge Identification Scheme

An identification scheme consists of three probabilistic, polynomial-time algorithms (G,P,V) such that:

- The randomized key generation algorithm G takes as input the security parameter 1^λ. It outputs a pair of keys $(\mathsf{pk},\mathsf{sk})$, where $\mathsf{pk}$is called the public key and sk is called the private key. We assume the security parameter is implicit in both pk and sk.
- P and V are interactive protocols. The prover algorithm P takes as input a private key sk and the verification algorithm V takes as input a public key pk. At the conclusion of the protocol, V outputs 1 or $\perp$.

It holds the following properties:

Completeness: $[P(\mathsf{sk}),V(\mathsf{pk})] = 1$.
Honest V always accepts honest P.

Soundness: $\Pr([P^*,V(\mathsf{pk})] = 1) = \mathsf{negl}(\lambda)$.
Cheating P^* (not knowing sk) is rejected with overwhelming probability.

Zero-knowledge: $[P(\mathsf{sk}),V^*(\mathsf{pk})] \approx [Sim,V^*(\mathsf{pk})]$.
Cheating V^* learns nothing about sk.

2.3 Digital Signature

A *digital signature scheme* Σ = (**Key Generation**, **Signature Generation**, **Signature Verification**) is consists of three algorithms.

Key Generation: The key generation algorithm takes a security parameter 1^λ and outputs a pair of keys $(\mathsf{pk}, \mathsf{sk})$.

Signature Generation: The signature generation algorithm takes a message m and a private key skas inputs and outputs a signature σ on the message m.

Signature Verification: The signature verification algorithm takes as input a public key pk, a message m and a signature σ, and outputs a bit denoting accept or reject, respectively.

The standard security notion for a signature scheme is *existential unforgeability against chosen message attack* (EUF-CMA): The forger gets a public key from a challenger who generates a key pair (sk, pk). The forger can query a signing oracle on polynomially many messages m_i hereby obtaining signatures σ_i. The forger can also issue a hash query and obtains its hash value. We say that the forger wins the EUF-CMA game, if the forger successfully outputs a pair (m^*,σ^*), where σ^* is a valid signature of a message m^* under the private key with the restriction that m^* has never appeared in the query phase.

2.4 Security Assumptions

An $[n,k]$ Linear Code $\mathcal{C}$ is a subspace of dimension k of the vector space $\mathbb{F}_q^n$. A linear code can be described by its Parity-check matrix H. The parity-check matrix describes the code as follows:

$$x \in \mathcal{C} \Leftrightarrow Hx = 0 \quad (\forall x \in \mathbb{F}_q^n).$$

[1]Jain et al. presented the ZK identification scheme based on the standard LPN problem as well, but it has higher soundness error as compared to the xLPN-based one. Hence it would result in signatures of larger size, and hence it is out of the scope of this work.

The product Hx is known as the *syndrome* of the vector x.

Definition 1. *Gilbert-Varshamov Bound:*
Let $\mathcal{C}$ be an $[n,k]$ linear code over $\mathbb{F}_q$. The Gilbert-Varshamov (GV) Distance is the largest integer ω such that

$$\frac{k}{n} = 1 - H_q\left(\frac{\omega}{n}\right),$$

where H_q is the q-ary entropy function

Now, we describe the main hard problems on which the security of code-based signature schemes, presented in the paper, relies.

Definition 2. *Syndrome Decoding Problem (SDP) (Augot et al., 2003):*

Input: *a matrix $H \in \mathbb{F}_2^{(n-k)\times n}$, a positive integer ω, and a vector $s \in \mathbb{F}_2^{n-k}$.*

Output: *a codeword x such that $\mathsf{wt}(x) = \omega'$ where $0 < \omega' \leq \omega$ and $Hx = s$.*

This problem is NP-complete (Berlekamp et al., 1978), which means that at least some instances of the problem are difficult. However, it is a common belief that they should be difficult on average (for well-chosen parameter ranges), which means that random instances are difficult. It is also proved that there exists a unique solution to SDP if the weight ω is below the GV Bound.

Definition 3. *General Decoding Problem (GDP):*

Input: *a matrix $G \in \mathbb{F}_2^{k\times n}$, a non negative integer ω, and a vector $y \in \mathbb{F}_2^n$.*

Output: *a pair $(m,e) \in \mathbb{F}_2^k \times \mathbb{F}_2^n$ such that $\mathsf{wt}(e) = \omega'$ where $0 < \omega' \leq \omega$ and $mG \oplus e = y$.*

An extension of SDP over arbitrary finite field is as follows:

Definition 4. *q-ary Syndrome Decoding Problem (q-SDP) (Augot et al., 2003):*

Input: *a matrix $H \in \mathbb{F}_q^{(n-k)\times n}$, a non negative integer ω and a vector $s \in \mathbb{F}_q^{n-k}$.*

Output: *a codeword x such that of $\mathsf{wt}(x) = \omega'$ where $0 < \omega' \leq \omega$ and $Hx = s$.*

q-SDP is proven to be NP-hard by S.Barg (Barg, 1994).

Definition 5. *Exact Learning Parity with Noise (xLPN) (Jain et al., 2012):*
The the decisional-xLPN problem is (n,t,ε)-hard if for every distinguisher D running in time t

$$\left| \Pr_{s,A,e}[D(A, As \oplus e) = 1] - \Pr_{A,r}[D(A,r) = 1] \right| \leq \varepsilon$$

where $s \leftarrow_{\$} \mathbb{F}_2^n$, $e \leftarrow_{\$} \{0,1\}^n$ such that $\mathsf{wt}(e) = \omega$, $r \leftarrow_{\$} \mathbb{F}_2^k$ and $A \leftarrow_{\$} \mathbb{F}_2^{k\times n}$.

3 SIGNATURE SCHEME

In this section, we will derive the signature scheme from code-based zero-knowledge identification schemes. Using Fiat-Shamir transformation (Pointcheval and Stern, 2000), we can derive signature schemes from the Stern (Stern, 1994) and Jain et al. (Jain et al., 2012) identification schemes. We need to use the extended version of Fiat-Shamir transformation (Alaoui et al., 2012) to derive a signature scheme from Cayrel et al.'s scheme (Cayrel et al., 2010).

3.1 Stern Signature Scheme.

System Parameters. The Stern signature scheme uses the following system parameters:

- Positive integer n (length of codeword),
- Positive integer k such that $k < n$ (dimension of the code),
- Positive integer ω (minimum distance of the code),
- Matrix $H \in \mathbb{F}_2^{(n-k)\times n}$ sampled randomly.
- random oracle: $h : \{0,1\}^* \to \{0,1\}^*$
- random oracle: $O : \{0,1\}^* \to \{0,1,2\}$

Key Generation. The key generation algorithm outputs the pair of the private key s and public key y.

1. Sample a vector $s \in \mathbb{F}_2^n$ such that $\mathsf{wt}(s) = \omega$.
2. Calculate a vector $y \in \mathbb{F}_2^{n-k}$ as $y = Hs$.
3. Output private key s and public key y.

Signature Generation. Takes as input private key s and a message m, and output Sig. The detailed algorithm is described in Algorithm 3.

Signature Verification. Takes as input public key y, message m, Sig. Compute $b_i \leftarrow O(m\|c_i)$ and output 1 if the following respective equation is valid for all $1 \leq i \leq \delta$:

$$\begin{cases} \text{Check } c_{i,0} = h(\sigma_i \| Hu_i) \text{ and } c_{i,1} = \sigma_i(u_i). & (b_i = 0) \\ \text{Check } c_{i,0} = h(\sigma_i \| H(u_i \oplus s) \oplus y) \text{ and} & \\ \quad c_{i,2} = h(\sigma_i(u_i \oplus s)). & (b_i = 1), \\ \text{Check} c_{i,1} = \sigma_i(u_i), c_{i,2} = h(\sigma_i(u_i) \oplus \sigma_i(s)), & \\ \quad \text{and } \mathsf{wt}(\sigma_i(s)) = \omega. & (b_i = 2) \end{cases}$$

or $\perp$ otherwise.

Algorithm 1: Signature Generation in Stern signature scheme.

Input: Private key s, Message m, and System parameters
Output: Signature Sig
for $i \leftarrow 0$ **to** $\delta - 1$ **do**
  $u_i \leftarrow_{\$} \mathbb{F}_2^n$;
  $\sigma_i \leftarrow_{\$} S_n$;
  $c_{i,0} \leftarrow h(\sigma_i \| Hu_i)$;
  $c_{i,1} \leftarrow \sigma_i(u_i)$;
  $c_{i,2} \leftarrow h(\sigma_i(u_i \oplus s))$;
  $c_i = (c_{i,0} \| c_{i,1} \| c_{i,2})$;
  $b_i = O(m \| c_i)$;
  $rsp_i \leftarrow \begin{cases} \sigma_i \| u_i & (b_i = 0) \\ \sigma_i \| (u_i \oplus s) & (b_i = 1) \\ \sigma_i(u_i) \| \sigma_i(s) & (b_i = 2) \end{cases}$;
  $sig_i = c_i \| rsp_i$;
end
$Sig \leftarrow sig_0 \| sig_1 \| \cdots \| sig_{\delta-1}$;
return Sig;

3.2 Jain et al. Signature Scheme

In this section, we will derive the signature scheme from the identification scheme of (Jain et al., 2012). We will modify the design a little, without any security breach, for the sake of fast implementation.

System Parameters. The signature scheme uses the system parameters as in Stern signature scheme apart from random sampling of $A \in \mathbb{F}_2^{k \times n}$ instead of $H \in \mathbb{F}_2^{(n-k) \times n}$.

Key Generation. The key generation algorithm outputs the pair of the private key e and public key y.

1. Sample $(s, e) \leftarrow_{\$} \mathbb{F}_2^k \times \mathbb{F}_2^n$ such that $\mathsf{wt}(e) = \omega$.
2. Calculate $y = sA \oplus e$.
3. Output private key e and public key y.

Signature Generation. The signature generation algorithm takes private key e and a message m, output Sig, and system parameters as inputs. Algorithm 2 describes the detailed algorithm.

Signature Verification. The signature verification algorithm takes as input public key y, message m, Sig, and system parameters. It computes $b_i \leftarrow O(m \| c_i)$ and outputs 1 if the following respective equation is valid for all $0 \leq i \leq \delta - 1$:

Algorithm 2: Signature Generation in Jain et al. signature scheme.

Input: Private key e, Message m, and System parameters
Output: Signature Sig
for $i \leftarrow 0$ **to** $\delta - 1$ **do**
  $u_i \leftarrow_{\$} \mathbb{F}_2^n$;
  $v_i \leftarrow_{\$} \mathbb{F}_2^k$;
  $\sigma_i \leftarrow_{\$} S_n$;
  $y_{i,0} \leftarrow v_i A \oplus u_i$;
  $c_{i,0} \leftarrow h(\sigma_i \| y_{i,0})$;
  $y_{i,1} \leftarrow \sigma_i(u_i)$;
  $c_{i,1} \leftarrow h(y_{i,1})$;
  $y_{i,2} \leftarrow \sigma_i(u_i \oplus e)$;
  $c_{i,2} \leftarrow h(y_{i,2})$;
  $c_i \leftarrow c_{i,0} \| c_{i,1} \| c_{i,2}$;
  $b_i \leftarrow O(m \| c_i)$;
  $rsp_i \leftarrow \begin{cases} \sigma_i \| y_{i,0} \| y_{i,1} & (b_i = 0) \\ \sigma_i \| y_{i,0} \| y_{i,2} & (b_i = 1) \\ y_{i,1} \| y_{i,2} & (b_i = 2) \end{cases}$;
  $sig_i \leftarrow c_i \| rsp_i$;
end
$Sig \leftarrow sig_0 \| sig_1 \| \cdots \| sig_{\delta-1}$;
return Sig;

$$\begin{cases} \text{Check } c_{i,0} \leftarrow h(\sigma_i \| y_{i,0}), & \\ \quad c_{i,1} \leftarrow h(y_{i,1}), & (b_i = 0) \\ \quad y_{i,0} \oplus \sigma_i^{-1}(y_{i,1}) = xA, \text{for some } x \text{ and } \sigma_i \in S_n & \\ \text{Check } c_{i,0} \leftarrow h(\sigma_i \| y_{i,0}), & \\ \quad c_{i,2} \leftarrow h(y_{i,2}), & (b_i = 1) \\ \quad \text{and } y_{i,0} \oplus \sigma_i^{-1}(y_{i,2}) \oplus y = xA, \text{for some } x & \\ \text{Check } c_{i,1} \leftarrow h(y_{i,1}), & \\ \quad c_{i,2} \leftarrow h(y_{i,2}), & (b_i = 2) \\ \quad \text{and } \mathsf{wt}(y_{1,i} \oplus y_{2,i}) = \omega & \end{cases},$$

or $\perp$ otherwise.

3.3 Cayrel et al. Signature Scheme

To present the Cayrel et al., signature scheme, first we introduce a special transformation that will be used in the scheme.

Definition 6. *Let $\sigma \in S_n$ and $\gamma = (\gamma_1, \cdots, \gamma_n) \in (\mathbb{F}_q^*)^n$ such that $\gamma_i \neq 0$ for all i. The transformation $\Pi_{\gamma,\sigma}$ is defined as follows:*

$$\Pi_{\gamma,\sigma} : \mathbb{F}_q^n \to \mathbb{F}_q^n$$
$$v \mapsto \left(\gamma_{\sigma[0]} v_{\sigma[0]}, \gamma_{\sigma[1]} v_{\sigma[1]}, \cdots, \gamma_{\sigma[n-1]} v_{\sigma[n-1]}\right).$$

Notice that this transformation is linear transformation, and satisfies $\Pi_{\gamma,\sigma}(v + w) = \Pi_{\gamma,\sigma}(v) + \Pi_{\gamma,\sigma}(w)$

and $\Pi_{\gamma,\sigma}(\alpha v) = \alpha\Pi_{\gamma,\sigma}(v)$ for all $v, w, \alpha \in \mathbb{F}_q$. Furthermore, the transformation preserves the Hamming weight of the vector.

Now, we are in the state to present the signature scheme:

System Parameters. The signature scheme uses the following system parameters:

- Positive integer n (length of codeword),
- Positive integer k such that $k < n$ (dimension of the code),
- Positive integer ω (minimum distance of the code),
- Matrix $H \in \mathbb{F}_q^{(n-k)\times n}$ sampled randomly,
- Random oracle $h : \{0,1\}^* \to \{0,1\}^*$,
- Random oracle $O_1 : \{0,1\}^* \to \mathbb{F}_q^n$,
- Random oracle $O_2 : \{0,1\}^* \to \{0,1\}$.

Key Generation. The key generation algorithm outputs the pair of the private key s and public key y.

1. Sample a vector $s \in \mathbb{F}_q^n$ such that $\mathsf{wt}(s) = \omega$.
2. Calculate a vector $y \in \mathbb{F}_q^{n-k}$ as $y = Hs$.
3. Output private key s and public key y.

Signature Generation. The signature generation algorithm takes as input private key s and a message m, and output a signature Sig. The detailed algorithm is described in Algorithm 3.

Signature Verification. The signature verification algorithm takes public key y, message m, and signature Sig as inputs. It computes $\alpha_i \leftarrow O_1(m\|c_i)$ and $b_i \leftarrow O_2(m\|c_i\|\alpha_i\|\beta_i)$; then, it outputs 1 if the following respective equation is valid for all $0 \le i \le \delta - 1$:

$$\begin{cases} \text{Check } c_{i,0} = h(\sigma_i\|\gamma_i\|H\Pi^{-1}_{\gamma_i,\sigma_i}(\beta_i) - \alpha_i y) & (b_i = 0) \\ \text{Check } c_{i,1} = h(\beta_i - \alpha_i\Pi_{\gamma_i,\sigma_i}(s)\|\Pi_{\gamma_i,\sigma_i}(s)) & \\ \quad \text{and } \mathsf{wt}(\Pi_{\gamma,\sigma}(s)) = \omega & (b_i = 1) \end{cases},$$

or $\perp$ otherwise.

Algorithm 3: Signature Generation in Cayrel et al. signature scheme.

Input: Private key s, Message m, and System parameters
Output: Signature Sig
for $i \leftarrow 0$ **to** $\delta - 1$ **do**
  $u_i \leftarrow_\$ \mathbb{F}_q^n$;
  $\sigma_i \leftarrow_\$ S_n$;
  $\gamma_i \leftarrow_\$ (\mathbb{F}_q \setminus \{0\})^n$;
  $c_{i,0} \leftarrow h(\sigma_i\|\gamma_i\|Hu_i)$;
  $c_{i,1} \leftarrow h(\Pi_{\gamma_i,\sigma_i}(u_i)\|\Pi_{\gamma_i,\sigma_i}(s))$;
  $c_i \leftarrow c_{i,0}\|c_{i,1}$;
  $\alpha_i \leftarrow O_1(m\|c_i)$;
  $\beta_i \leftarrow \Pi_{\gamma_i,\sigma_i}(u_i + \alpha_i s)$;
  $b_i \leftarrow O_2(m\|c_i\|\alpha_i\|\beta_i)$;
  $rsp_i \leftarrow \begin{cases} \sigma_i\|\gamma_i & (b_i = 0) \\ \Pi_{\gamma_i,\sigma_i}(s) & (b_i = 1) \end{cases}$;
  $sig_i = c_i\|\beta_i\|rsp_i$;
end
$Sig \leftarrow sig_0\|sig_1\|\cdots\|sig_{\delta-1}$;
return Sig;

4 SECURITY ANALYSIS AND PARAMETER SELECTION

Best Known Attack: It is required to consider *structural attack and* key-recovery attack to measure the security of code-based signature schemes. Due to the use of random code only, consideration of structural attack abolish. Signature schemes, constructed by using Fiat-Shamir transformation and it is extended version on zero knowledge identification schemes, are EUF-CMA secure (Pointcheval and Stern, 2000; Alaoui et al., 2012). Moreover, EUF-CMA security includes the security against key-recovery attack. Therefore, to choose secure parameters, it is required to measure the hardness of the problem, where the proof of EUF-CMA security is reduced.

The security of the Stern and Cayrel et al. signatures are reduced to the hardness of SDP and qSDP, respectively. The most efficient known algorithm to attack SDP is the Information Set Decoding (ISD) algorithm by Stern (Stern, 1988). Further, there are few many improvements over (Stern, 1988). One of the intermediate notable improvement is by Finiasz et al. (Finiasz and Sendrier, 2009). Further improvements over (Finiasz and Sendrier, 2009) are asymptotic (May et al., 2011; Becker et al., 2012). So, we have used the measurement of (Finiasz and Sendrier, 2009) to measure the hardness of SDP. The hardness of q-ary SDP is measured by the formulation of (Niebuhr et al., 2017), which is the extension of (Finiasz and Sendrier, 2009) from binary to an arbitrary finite field.

Security of Jain et al. signature scheme is reduced to the hardness of xLPN. However, in actuality, the number of sample of the instance is small, and we cannot apply the BKW algorithm to solve the xLPN problem. So, the hardness is turned down to the hard-

Table 1: System parameters and data sizes for Stern signature scheme.

Parameter	80-bit Security	128-bit Security
Independent parameters:		
n	620	1,024
δ	137	219
Derived parameters:		
k	310	512
ω	68	112
Data size:		
sk	620 bit	1024 bit
pk	310 bit	512 bit
Signature	93.3 kB	245 kB
Systemf param.	24.0 kB	65.5 kB

ness of the general decoding problem. We thus have used the measurement of (Finiasz and Sendrier, 2009) to measure the hardness.

4.1 Stern Signature Scheme

We select parameters n, k and ω so that they lie on the GV bound i.e.,

$$\frac{k}{n} = 1 - H_2\left(\frac{\omega}{n}\right), \tag{1}$$

to maximize the security against attacks using the ISD algorithm. We select $k = n/2$ and ω is around $0.110n$. The complexity of the ISD (Finiasz and Sendrier, 2009) is

$$WF_{\mathsf{ISD}}(n,k,\omega) = \min_{p} \frac{2l \min\left(\binom{n}{\omega}, 2^r\right)}{(1-e^{-1})\binom{r-l}{\omega-p}\sqrt{\binom{k+l}{p}}} \tag{2}$$

where

$$l = \log_2\left(2\omega\sqrt{\binom{k}{p}}\right)$$

for the parameters on the GV bound, and n should satisfy $WF_{\mathsf{ISD}}(n,k,\omega) > 2^{\lambda}$.

The number of rounds δ depends on a *soundness error* of the underlying identification scheme. The soundness error of the Stern identification scheme (Stern, 1994) is $2/3$. So, δ should satisfy $(2/3)^{\delta} < 2^{-\lambda}$. System parameters and data sizes are presented in Table 1.

4.2 Jain et al. Signature Scheme

We select parameters n, k and ω according to the equation 1 & 2. The number of rounds δ depends on the *Soundness error* of the underlying identification scheme. Soundness error of the Jain et al. identification scheme (Jain et al., 2012) is $2/3$. So, δ should satisfy $(2/3)^{\delta} < 2^{-\lambda}$. System parameters and data sizes are presented in Table 2.

Table 2: System parameters and data sizes of Jain et al. signature scheme.

Name	80-bit Security	128-bit Security
Independent parameters:		
n	620	1,024
δ	137	219
Derived parameters:		
k	310	512
ω	68	112
Data size:		
sk	930 bit	1536 bit
pk	620 bit	1024 bit
Signature	95.11 kB	263 kB
System param.	24.0 kB	65.5 kB

4.3 Cayrel et al. Signature Scheme

We select parameters n, k and ω so that they lie on the GV bound i.e.,

$$\frac{k}{n} = 1 - H_q\left(\frac{\omega}{n}\right),$$

to maximize the security against attacks using the ISD algorithm. We select $k = n/2$ and ω is around $0.380n$.

The relevant formula to evaluate work factor is as follows (Niebuhr et al., 2017):

$$\begin{aligned} &WF_{\mathsf{qISD}}(n,k,\omega,q) \\ &= \min_{l,p_1,p_2} \frac{N_{p,q}(l)}{\sqrt{q-1}}\left(\lambda_q^{-1}\left(\frac{2(q-1)l}{(q\binom{l}{p_2'})(q-1)p_2'} + p_2\right)\right. \\ &\times \left.\sqrt{\binom{k}{p_1}\binom{l}{p_2}}(q-1)^{p-1} + K_q\frac{\binom{k}{p_1}\binom{l}{p_2}(q-1)^{p-1}}{q^l}\right) \end{aligned}$$

where

$$N_{p,q}(l) = \frac{\min(\binom{n}{\omega}(q-1)^{\omega}, q^{n-k})}{\binom{n-k-l}{\omega-p}\binom{k}{p_1}\binom{i}{p_2}(q-1)^{\omega}},$$

$p = p_1 + p_2$, $p_2' = \lfloor p_2/2 \rfloor$, $\lambda_q = 1 - e^{-1} \approx 0.63$ and $\lfloor p_2/2 \rfloor$ means maximum integer that does not exceed $p_2/2$. n should satisfy $WF_{\mathsf{ISD}}(n,k,\omega) > 2^{\lambda}$.

The number of rounds δ depends on the *Soundness error* of the underlying identification scheme. Soundness error of the Cayrel et al. identification scheme (Cayrel et al., 2010) is $1/2$. So, δ should satisfy $(1/2)^{\delta} < 2^{-\lambda}$. System parameters and data sizes are presented in Table 3.

5 IMPLEMENTATIONS

We implement the Stern, Jain et al. and Cayrel et al. signature schemes with 128-bit level security in C language. Execution time and data sizes are presented in

Table 3: System parameters and data sizes for Cayrel et al. signature scheme over $\mathbb{F}_{256}$.

Parameter	80-bit Security	128-bit Security
Independent parameters:		
n	144	230
δ	80	128
Derived parameters:		
k	72	115
ω	54	87
Data size:		
sk	1,152 bits	1,840 bits
pk	576 bits	920 bits
Signature	89.6 kB	229 kB
System param.	10.4 kB	26.5 kB

Table 4: Execution time and data sizes of the three signature schemes for 128-bit security level.

	Stern	Jain et al.	Cayrel et al.
Key-gen	0.0170 ms	0.0201 ms	0.339 ms
Sign	31.5 ms	16.5 ms	24.3 ms
Verify	2.27 ms	135 ms	9.81 ms
sk	1024 bit	1536 bit	1840 bit
pk	512 bit (+ 65.5 kB)*	512 bit (+ 65.5 kB)*	920 bit (+ 229 kB)*
Signature	245 kB	263 kB	229 kB

* System parameters

Table 4. Eight variables of $\mathbb{F}_2$ in the Stern and Jain et al. signature schemes, and a variable of $\mathbb{F}_{256}$ in the Cayrel signature scheme are stored in an eight-bit `uint8` variable. Our implementation uses a pre-computation table for multiplication between $\mathbb{F}_{256}$ elements. We used SHA3-256 to implement random oracles used in the signature schemes. For example, a random oracle $h : \{0,1\}^* \to \{0,1\}^{1024}$ is implemented as $v \mapsto$ SHA3-256($\texttt{0x00}\|v$)$\|$SHA3-256($\texttt{0x01}\|v$)$\|$ SHA3-256($\texttt{0x02}\|v$)$\|$SHA3-256($\texttt{0x03}\|v$). One byte prefix `0x00`, `0x01`, ..., `0x03` are auxiliary inputs to achieve independent hash functions. Durstenfeld Shuffle (Durstenfeld, 1964) that outputs a random permutation within $O(n)$ computational complexity, is used in the signature schemes.

The signature size in the Cayrel et al. signature scheme is smaller and the execution time of the signature generation algorithm is smaller comparing to the Stern signature scheme. Conversely, the signature verification algorithm in the Stern signature scheme is faster than that in the Cayrel et al. signature scheme since it consists only of hash calculations, permutations, exclusive-or operations, and Hamming weight checks. Table 4 show the execution time of the signature schemes on a PC with a 3.5 GHz CPU and 16 GB of RAM and the size of the secret key, public key, and a signature. The size of the input messages is 32 B.

Alaoui et al. (Alaoui et al., 2013), implemented the signature schemes of Stern, Veron and Cayrel et al. for 80 bits of security. Further, security of Veron's scheme is fixed by Jain et al. We have implemented the signature schemes of the Stern, Jain et al. and Cayrel et al. for 128 bits of security. In table 5, we have provided a comparison with the Stern and Cayrel et al. schemes.

Table 5: Comparison between our Implementation and (Alaoui et al., 2013).

	Stern		Cayrel et al.	
	Sign (ms)	Verific. (ms)	Sign (ms)	Verific. (ms)
Our Impliment. (128 bits of sec.)	31.5	2.27	24.3	9.81
(Alaoui et al., 2013) (80 bits of sec.)	7.18	3.57	4.25	1.90

6 CONCLUSION

We derived code-based signature schemes using Fiat-Shamir transformation on code-based zero-knowledge identification schemes: the Stern scheme, the Jain-Krenn-Pietrzak-Tentes scheme, and the Cayrel-Veron-Alaoui scheme. We then analyzed the security of the three signature schemes and derived the security parameters to achieve the 128-bit level of security. Furthermore, we implemented these signature schemes and compared their performance on a PC. Our security analysis and implementation show the Stern signature scheme is the most efficient regarding the size of the secret and public key and the execution time of key generation and signature verification, and the Cayrel et al. scheme is the most superior in terms of signature size. The signature generation of the Jain et al. scheme is the fastest, but the signature verification of the scheme is slowest. In our future work, we will optimize the implementation for IoT devices and study techniques for reducing the signature size.

REFERENCES

Alaoui, S. M. E. Y., Cayrel, P.-L., Bansarkhani, R. E., and Hoffmann, G. (2013). Code-Based Identification and Signature Schemes in Software. In *CD-ARES Workshops 2013*, pages 122–136.

Alaoui, S. M. E. Y., Dagdelen, Ö., Véron, P., Galindo, D., and Cayrel, P.-L. (2012). Extended security arguments for signature schemes. In *International Conference on Cryptology in Africa*, pages 19–34. Springer.

Augot, D., Finiasz, M., and Sendrier, N. (2003). A Fast Provably Secure Cryptographic Hash Function. Cryptology ePrint Archive, Report 2003/230. https://eprint.iacr.org/2003/230.

Barg, S. (1994). Some new np-complete coding problems. *Problemy Peredachi Informatsii*, 30(3):23–28.

Becker, A., Joux, A., May, A., and Meurer, A. (2012). Decoding random binary linear codes in 2n/20: How 1+ 1= 0 improves information set decoding. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 520–536. Springer.

Berlekamp, E., McEliece, R., and van Tilborg, H. (1978). On the Inherent Intractability of Certain Coding Problems. *IEEE Transactions on Information Theory*, 24(3):384–386.

Cayrel, P.-L., Gaborit, P., and Prouff, E. (2008). Secure implementation of the stern authentication and signature schemes for low-resource devices. In *International Conference on Smart Card Research and Advanced Applications*, pages 191–205. Springer.

Cayrel, P.-L., Véron, P., and Alaoui, S. M. E. Y. (2010). A zero-knowledge identification scheme based on the q-ary syndrome decoding problem. In *International Workshop on Selected Areas in Cryptography*, pages 171–186. Springer.

Courtois, N. T., Finiasz, M., and Sendrier, N. (2001). How to Achieve a McEliece-Based Digital Signature Scheme. In *Advances in Cryptology – ASIACRYPT 2001. ASIACRYPT 2001. Lecture Notes in Computer Science*, volume 2248, pages 157–174.

Dallot, L. (2008). Towards a Concrete Security Proof of Courtois, Finiasz and Sendrier Signature Scheme. In *Research in Cryptology. WEWoRC 2007. Lecture Notes in Computer Science*, volume 4945, pages 65–77.

Debris-Alazard, T., Sendrier, N., and Tillich, J.-P. (2017). The problem with the SURF scheme. https://arxiv.org/abs/1706.08065v4.

Durstenfeld, R. (1964). Algorithm 235: Random permutation. *Commun. ACM*, 7(7):420–.

Faugére, J.-C., Gauthier-Umana, V., Otmani, A., Perret, L., and Tillich, J.-P. (2011). A Distinguisher for High-Rate McEliece Cryptosystems. In *Information Theory Workshop (ITW), IEEE*, pages 282–286.

Finiasz, M. and Sendrier, N. (2009). Security Bounds for the Design of Code-Based Cryptosystems. In *Advances in Cryptology – ASIACRYPT 2009*, volume 5912, pages 88–105.

Gaborit, P. and Girault, M. (2007). Lightweight code-based identification and signature. In *Information Theory, 2007. ISIT 2007. IEEE International Symposium on*, pages 191–195. IEEE.

Jain, A., Krenn, S., Pietrzak, K., and Tentes, A. (2012). Commitments and efficient zero-knowledge proofs from learning parity with noise. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 663–680. Springer.

Kabatianskii, G., Krouk, E., and Smeets, B. (1997). A Digital Signature Scheme Based on Random Error-Correcting Codes. In *Proceedings of the 6th IMA International Conference on Cryptography and Coding*.

May, A., Meurer, A., and Thomae, E. (2011). Decoding random linear codes in $\tilde{O}(2^{0.054n})$. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 107–124. Springer.

Morozov, K., Roy, P. S., Steinwandt, R., and Xu, R. (2018). On the security of the courtois-finiasz-sendrier signature. *Open Mathematics*, 16(1):161–167.

Niebuhr, R., Persichetti, E., Cayrel, P.-L., Bulygin, S., and Buchmann, J. (2017). On lower bounds for information set decoding over 𝕗 q and on the effect of partial knowledge. *International journal of information and Coding Theory*, 4(1):47–78.

Otmani, A. and Tillich, J.-P. (2011). An Efficient Attack on All Concrete KKS Proposals. In *Post-Quantum Cryptography. PQCrypto 2011. Lecture Notes in Computer Science*, volume 7071, pages 98–116.

Pointcheval, D. and Stern, J. (2000). Security Arguments for Digital Signatures and Blind Signatures. *Journal of Cryptology*, 13(3):361–396.

Shor, P. (1994). Polynomial-time algorithm for prime factorization and discrete logarithms on a quantum computer: proc. In *35th Annual Symp. on the Foundations of Computer Science*, volume 124.

Stern, J. (1988). A method for finding codewords of small weight. In *International Colloquium on Coding Theory and Applications*, pages 106–113. Springer.

Stern, J. (1994). A new identification scheme based on syndrome decoding. In *Advances in Cryptology – CRYPTO 1993. Lecture Notes in Computer Science*, volume 773, pages 13–21.

Véron, P. (1997). Improved identification schemes based on error-correcting codes. *Applicable Algebra in Engineering, Communication and Computing*, 8(1):57–69.