Bulletin of the *Transilvania* University of Braşov • Vol 2(51) - 2009
Series III: Mathematics, Informatics, Physics, 3-10

NEW RESULTS ON THE STERN IDENTIFICATION AND SIGNATURE SCHEME

Pierre-Louis CAYREL[1]

Communicated to:
9-ème Colloque franco-roumain de math. appl., 28 août-2 sept. 2008, Braşov, Romania

Abstract

In this paper, I propose a survey of new results on the Stern identification and signature scheme. I describe coding theory, its application to cryptography and different identification and signature schemes using error correcting codes. Next I briefly describe a secure implementation of the Stern scheme against DPA and an identity based signature scheme.

1 Introduction

Coding theory deals with the error-prone process of transmitting data accross noisy channels, via clever means, so that a large number of errors that occur can be corrected. It also deals with the properties of codes, and thus with their fitness for a specific application. We can use the coding theory in cryptography.

It is partly on the difficulty of the Syndrome Decoding problem and the Goppa Parameterized Bounded Decoding problem that the safety of several cryptosystems (e.g. McEliece [4] or Niederreiter) lies. This makes it an issue for cryptographic applications of error-correcting codes is not only being NP-complete, but also the fact that, in general, a random instance of this problem is always difficult. It is difficult to prove specific results on this matter, but it seems that in terms of complexity *in average*, the problem is almost always difficult.

Code-based cryptography was introduced by McEliece [4], two years after the introduction of public key cryptography by Diffie and Hellman in 1976. In his paper he proposed a public key encryption scheme. In 1986 Niederreiter proposed an equivalent code-based cryptosystem.

McEliece first had the idea in 1978, of using the theory of error-correcting codes for cryptographic purposes, and more specifically for asymmetric encryption algorithm. The principle of the protocol described is for Alice to send a message containing a large number of errors, errors that only Bob knows how to detect and correct.

[1]Université de Limoges, XLIM-DMI, 123, Av. Albert Thomas 87000 Limoges, France

2 Identification and signature with error correcting codes

An open question was the existence of a signature scheme based on coding theory, if one puts aside the signature scheme induced by the Fiat-Shamir paradigm from the Stern identification scheme of 1993, the first signature scheme was proposed in 2001 by Courtois, Finiasz and Sendrier in [3]. It adapts the *Full Domain Hash* approach of Bellare and Rogaway to Niederreiter's encryption scheme.

Stern Identification and Signature Scheme At Crypto'93, Stern proposed a new identification and signature scheme based on coding theory [5]. Stern's Scheme is an interactive zero-knowledge protocol which aims at enabling any user (usually called *the prover* and denoted here by P) to identify himself to another user (usually called *the verifier* and denoted here by V).

Let n and k be two integers such that $n \geq k$. Stern's scheme assumes the existence of a public $(n-k) \times n$ matrix $\widetilde{H}$ defined over $\mathbb{F}_2$. It also assumes that an integer $t \leq n$ has been chosen. For security reasons (discussed in [5]) it is recommended that t is chosen slightly below the value given by the Gilbert-Varshamov bound. The matrix $\widetilde{H}$ and the weight t are protocol parameters and may be used by several (even numerous) different provers.

Each prover P receives a n-bit secret key s_P (also denoted by s if there is no ambiguity about the prover) of Hamming weight t and computes a *public identifier* i_V such that $i_V = \widetilde{H} s_P^T$. This identifier is calculated once for each $\widetilde{H}$ and can thus be used for several identifications. When a user P needs to prove to V that he is indeed the person associated with the public identifier i_V, then the two protagonists perform the protocol described in the Figure 2 where $h(a\|b)$ denotes the hash of the concatenation of the sequences a and b.

1) [Commitment Step] P randomly chooses $y \in \mathbb{F}^n$ and a permutation σ defined over $\mathbb{F}_2^n$. Then P sends to V the commitments c_1, c_2 and c_3 such that :

$$c_1 = h(\sigma \| \widetilde{H} y^T);\ c_2 = h(\sigma(y));\ c_3 = h(\sigma(y \oplus s)),$$

2) [Challenge Step] V sends $b \in \{0, 1, 2\}$ to P.

3) [Answer Step] Three possibilities: P reveals

$$\text{if } b = 0 : y \text{ and } \sigma;\ \text{if } b = 1 : (y \oplus s) \text{ and } \sigma;\ \text{if } b = 2 : \sigma(y) \text{ and } \sigma(s).$$

4) [Verification Step] Three possibilities : V checks :

$$\text{if } b = 0 : c_1, c_2;\ \text{if } b = 1 : c_1, c_3;\ \text{if } b = 2 : c_2, c_3 \text{ and } wt(\sigma(s)) = t.$$

5)Iterate the steps 1,2,3,4 until the expected security level is reached.

Figure 1: Stern's Protocol

As proved in [5], the protocol is zero-knowledge and for a round iteration, the probability that a dishonest person succeeds in cheating is (2/3). Therefore, to get a confidence level of β, the protocol must be iterated a number of times k such that $(2/3)^k \leq \beta$ holds.

When the number of iterations satisfies the last condition, then the security of the scheme relies on the NP complete problem SD.

By using the so-called Fiat-Shamir paradigm, it is theoretically possible to convert Stern's Protocol into a signature scheme, but then the signature is very long (about 140-kbit long for a security of 2^{80} binary operations).

CFS signature's scheme Courtois, Finiasz and Sendrier proposed in [3] a practical signature scheme based on coding theory (CFS). The idea of the CFS scheme is to fix parameters $[n, k, d]$ such that the density of decodable codewords is reasonable and pick up random elements until one is able to decode it. More precisely, given M a message to sign and h a hash function of $\{0,1\}^{n-k}$. We try to find a way to build $s \in \mathbb{F}_2^n$ of given weight t such that $h(M) = \widetilde{H}s^T$. For $D()$ a decoding algorithm, the algorithm works as follows :

1) $i \leftarrow 0$
2) while $h(M\|i)$ is not decodable do $i \leftarrow i + 1$
3) compute $s = D(h(M\|i))$

Figure 2: The CFS signature scheme

We get a couple $\{s, j\}$, such that $h(M \oplus j) = \widetilde{H}s^T$. Let us notice that we can suppose that s has weight $t = [d/2]$. Using a Goppa code, we start from a word of length $k = n - mt$ that we transform into a codeword of length $n = 2m$, and then we have an error of weight t. The decoding algorithm (the trapdoor) permits us to decode any word built that way, meaning, the set of all words to a distance t or less of a codeword.

3 Secure implementation of the Stern scheme

In [2], the authors present the first implementation on smartcard of the Stern identification and signature scheme. This gives a securization of the scheme against side channel attacks. On the whole, this provides a secure implementation of a very practical identification (and possibly signature) scheme which is mostly attractive for light-weight cryptography.

A quick analysis of Stern's protocol shows that the different steps are merely composed of the four following main operators:
1) - Matrix-vector product: a vector and a random double circulant matrix;
2) - Hash function: the action of a hash function;
3) - Permutation: the generation and the action of a random permutation on words;
4) - PRNG: to generate random permutations and vectors.

Side-channel attacks aim at recovering information about *sensitive variable* appearing in the description of the algorithm under attack. We shall say that a variable is sensitive if it is a function of public data and of a secret (resp. private) parameter of the algorithm

(the function must be not constant with respect to the secret parameter).

After a brief listing of the variables involved, we get the following list of sensitive variables that can be potentially targeted by an SCA:

Threat A. the random vector y in the computations of c_1 (when performing $\tilde{H}y^T$) and of c_2 (when performing $\sigma(y)$): if an attacker is able to retrieve y during one of these steps, then with a probability 1/3 he is able to recover s when A answers $y \oplus s$ to the $(b = 1)$-challenge.

Threat B. the private vector s during the computation $\sigma(s)$: in this case the attacker recovers A's private parameter.

Threat C. the private vector s during the computation $\sigma(y \oplus s)$: in this case the attacker recovers A's private parameter.

Threat D. the bit-permutation σ during the computation of $\sigma(s)$ or $\sigma(y \oplus s)$: if an attacker is able to retrieve σ during one of these steps, then with a probability 1/3 he is able to recover s when A answers $\sigma(s)$ to the $(b = 2)$-challenge.

Threat E. the bit-permutation σ during the computation of the hash value $h(\sigma \| \tilde{H}y^T)$: if an attacker is able to retrieve σ, then with a probability 1/3 he is able to recover s when A answers $\sigma(s)$ to the $(b = 2)$-challenge.

Three categories of side-channel attacks can be defined, depending on their modus operandi.

1. The so-called *template* attacks consist in a long off-line profiling step, that enables future fast on-line attacks.

2. The so-called *simple analysis* attacks (SPA in short) are on-line attacks that consist in directly interpreting power consumption measurements and in identifying the execution sequence.

3. The so-called *correlation* attacks work as *greedy* algorithms: the side-channel information is analyzed until the secrets are extracted.

The securisation is based on the fact that the linear operations (scalar products or bit-permutations) are easy to implement in hardware and are very efficient to secure against DPA attacks. We obtain an identification in 6 seconds and and a signature in 24 seconds for a security of 2^{85}. The communication cost is around 40-kbits in the identification scheme and around 140-kbits for the signature.

For a satisfying security level, the size of the public key is only 694 bits using a quasi cyclic representation of the matrix considered. The double-circulant matrices are a good trade-off between random and strongly structured matrices. In this case the operations are indeed really simple to perform and can be implemented easily in hardware. Moreover, the fact that the protocol essentially performs linear operations makes the algorithm easy to protect against side channel attacks. We thus think that the protocol is a new option to

carry out fast strong identification on smart cards. Additionally, we think that the use of a dedicated linear-algebra co-processor should significantly improve the timing performances of our implementation.

Future work besides this one includes considering Fault injection attacks and implementation of other variations of Stern protocol which can have other small advantages.

4 Identity-based signature scheme

In [1], the authors present an IBI (identity based identification) and an IBS (identity based signature) scheme based on error-correcting code. Here is the scheme :

Given C a q-ary linear code of length n and of dimension k. Let H be a parity check matrix of C. Given $\widetilde{H} = VHP$ with V invertible and P a matrix of permutation. Let h a hash function with values in $\{0,1\}^{n-k}$. Let id_A Alice's identity, id_A can be compute by everyone. Similarly, $\widetilde{H}$ is public. The decomposition of $\widetilde{H}$ is, on the contrary, a secret of the authority and not of Alice.
We shall describe an identity-based identification method : Alice the prover is identifying herself to Bob the verifier.

Preliminary : key deliverance
Alice has to authentify herself in a classic way, to get the private key which will then allow her to authentify herself to a third person as Bob. For that purpose, we use variation on identity.

Let us admit that we know Alice's identity id_A. Given h a hash function with values in $\{0,1\}^{n-k}$. We search a way to find $s \in E_{q,n,t}$ such that $h(id_A) = \widetilde{H}s^T$.
The main point is to decode $h(id_A)$. The main problem is that $h(id_A)$ is not *in principle* in the arrival space of $x \to \widetilde{H}x^T$. That is to say that $h(id_A)$ is not *in principle* in the space of decodable elements of $\mathbb{F}_2^n$. That problem can be solved thanks to the following algorithm. Given $D()$ a decoding algorithm for the hidden code:

1. $i \leftarrow 0$
2. while $h(id_A \| i)$ is not decodable do $i \leftarrow i + 1$
3. compute $s = D(h(id_A \| i))$
   where $arg_1 \| arg_2$ notes the concatenation of arg_1 and arg_2.

Figure 3: key deliverance

We get at the end a couple $\{s, j\}$, such that $h(id_A \| j) = \widetilde{H}s^T$. We can note that we have s of weight t or less.

Identification by Bob.
We use a slight derivation of Stern's protocol. We suppose that A obtained a couple $\{s, j\}$ verifying $h(id_A \| j) = \widetilde{H}s^T$. $h(id_A \| j)$ is A's public key. The new protocol is based

on Stern's protocol but with two changes (steps 1 and 4, see figure 3, we keep the same notations)

The security of this system is the same as the security of Stern's one. This scheme is the first proposed non number theory based identity based scheme.

The knowledge of j doesn't permit to find s such that $h(id_A \| j) = \widetilde{H} s^T$. The security of this system is the same as the security of Stern's one.

It is possible to derive a signature scheme from the zero-knowledge identification scheme of Stern by classical constructions. Hence it permits to derive an IBS scheme.

1. [Commitment Step] P randomly chooses $y \in \mathbb{F}^n$ and a permutation σ defined over $\mathbb{F}_2^n$. Then P sends to V the commitments c_1, c_2 and c_3 such that :
$$c_1 = h(\sigma \| \widetilde{H} y^T);\ c_2 = h(\sigma(y));\ c_3 = h(\sigma(y \oplus s)).$$
2. [Challenge Step] V sends $b \in \{0, 1, 2\}$ to P.
3. [Answer Step] Three possibilities:
   - if $b = 0 : P$ reveals y and σ.
   - if $b = 1 : P$ reveals $(y \oplus s)$ and σ.
   - if $b = 2 : P$ reveals $\sigma(y)$ and $\sigma(s)$.
4. [Verification Step] Three possibilities:
   - if $b = 0 : V$ verifies that c_1, c_2 are correct.
   - if $b = 1 : V$ verifies that c_1, c_3 are correct.
   - if $b = 2 : V$ verifies that c_2, c_3 are correct, and that the weight of $\sigma(s)$ is t.
5. Iterate the steps 1,2,3,4 until the expected security level is reached.

Figure 4: Stern's Protocol (identity-based version)

5 Conclusion

Recently a few improvements have been done in code based signature schemes. The reduction of the size of the public key is a real breakthrought for an use in low ressource context [2]. We have shown in this paper that the Stern identification and signature scheme could be interesting in different areas. Besides this, the Stern scheme has also the

3 following advantages :

1. can be an alternative to the number-theory based protocols;
2. just based on linear operations (scalar products or bit-permutations);
3. the secret key is smaller than the one of the other protocols (a few hundred bits) for the same security level.

References

[1] Cayrel, P.-L., Gaborit P., Girault, M., *Identity-based identification and signature schemes using correcting codes*, In D. Augot, N. Sendrier, and J.-P. Tillich, editors, WCC 2007. INRIA, 2007.

[2] Cayrel, P.-L., Gaborit P., Prouff, E., *Secure implementation of the Stern authentication and signature schemes for low-resource devices*, CARDIS, 2008.

[3] Courtois, N., Finiasz M., Sendrier, N., *How to achieve a Mceliece based digital signature scheme*, Springer-Verlag, 2001.

[4] McEliece, R. J., *A public-key cryptosystem based on algebraic coding theory*, JPL DSN Progress Report, 114-116, 1978.

[5] Stern, J., *A new identification scheme based on syndrome decoding*, In D. Stinson, editor, Advances in Cryptology – CRYPTO'93, volume 773 of LNCS, 13-21, Springer-Verlag, 1993.