Side channels attacks in code-based cryptography

Pierre-Louis Cayrel[1] and Falko Strenzke[2]

1 - CASED – Center for Advanced Security Research Darmstadt,
Mornewegstrasse, 32 64293 Darmstadt Germany
pierre-louis.cayrel@cased.de
2 - FlexSecure GmbH, Germany,
strenzke@flexsecure.de

Abstract. The McEliece and the Niederreiter public key cryptosystems (PKC) are supposed secure in a post quantum world [4] because there is no efficient quantum algorithm for the underlying problems upon which these cryptosystems are built. The CFS, Stern and KKS signature schemes are post-quantum secure because they are based on hard problems of coding theory. The purpose of this article is to describe what kind of attacks have been proposed against code-based constructions and what is missing.

Keywords: post-quantum cryptography, code-based cryptography, side-channel attacks.

1 Introduction

In 1978, R. J. McEliece presented the first version of the cryptosystem which was to become the reference in public key cryptography based on coding theory [7]. The main version of McEliece's scheme uses Goppa codes. However, many other code families have been studied to fit in McEliece's system.

The McEliece cryptosystem presents a good alternative to classic number theory encryption schemes. Its security has been studied for years and a lot of distinct constructions have been proposed in order to reduce the huge size of public keys. After several improvements, the use of binary quasi-cyclic alternant codes leads to a public key of 6,500 bits (see [3]) and 4,096 bits using binary quasi-dyadic Goppa codes [2]. These two constructions seem secure against decoding and structural attacks.

In the real world, it's important to study side channel attacks against hardware implementations. A first study of the McEliece scheme against side channel attacks has been done in [8]. An other study of a code-based scheme has been proposed for the Stern identification and signature scheme [5].

Our contribution

We present here the state of the art of side-channels attacks against code-based cryptosystems which is *not very substantial* in order to show that there remains a lot of work to do in this area.

2 Preliminaries

2.1 The McEliece PKC

The McEliece PKC [7] represents one of the oldest public-key cryptosystem ever designed. It is also the first public-key cryptosystem based on linear error-correcting codes. The principle is to select a linear code of length n and dimension t that is able to efficiently correct t errors. The core idea is to transform it to a random-looking linear code. A description of the original code and the transformations can serve as the private key while a description of the modified code serves as the public key. McEliece's original proposal uses a generator matrix of a binary Goppa code. The encryption function encodes a message according to the public code and adds an error vector of weight t. The decryption function basically decodes the ciphertext by recovering the secret code through the trapdoor which consists of the transformation between the public and the private code and the Patterson algorithm for the binary Goppa codes that makes use of the secret Goppa polynomial.

The McEliece cryptosystem [7] uses error-correcting codes that have an efficient decoding algorithm in order to build trapdoor one-way functions. McEliece proposed binary Goppa codes as the underlying family of codes. Figure 1 and Figure 2 give details of the three algorithms.

Fig. 1. Key generation algorithm of the McEliece cryptosystem

```
KeyGen($1^\kappa$) ($\kappa$ is the security parameter)
1. Choose $n$, $k$ and $t$ according to $\kappa$
2. Randomly pick a generator matrix $\boldsymbol{G}_0$ of an $[n, k, 2t + 1]$ binary Goppa code $\mathscr{C}$
3. Randomly pick a $n \times n$ permutation matrix $\boldsymbol{P}$
4. Randomly pick a $k \times k$ invertible matrix $\boldsymbol{S}$
5. Calculate $\boldsymbol{G} = \boldsymbol{S} \times \boldsymbol{G}_0 \times \boldsymbol{P}$
6. Output $\mathsf{pk} = (\boldsymbol{G}, t)$ and $\mathsf{sk} = (\boldsymbol{S}, \boldsymbol{G}_0, \boldsymbol{P}, \gamma)$ where $\gamma$ is a $t$-bounded decoding algorithm of $\mathscr{C}$
```

Fig. 2. Encryption and decryption algorithms of the McEliece cryptosystem

```
Encrypt(pk,$\boldsymbol{m} \in \mathbb{F}_2^k$)
1. Randomly pick $\boldsymbol{e}$ in $\mathbb{F}_2$ of weight $t$
2. Calculate $\boldsymbol{c} = \boldsymbol{m} \times \boldsymbol{G} + \boldsymbol{e}$
3. Output $\boldsymbol{c}$
```

```
Decrypt(sk,$\boldsymbol{c} \in \mathbb{F}_2^n$)
1. Calculate $\boldsymbol{z} = \boldsymbol{c} \times \boldsymbol{P}^{-1}$
2. Calculate $\boldsymbol{y} = \gamma(\boldsymbol{z})$
3. Output $\boldsymbol{m} = \boldsymbol{y} \times \boldsymbol{S}^{-1}$
```

2.2 Niederreiter Cryptosystem

A dual encryption scheme is the Niederreiter cryptosystem [9] which is equivalent in terms of security [6] to the McEliece cryptosystem. The main difference between McEliece and Niederreiter cryptosystems lies in the description of the codes. The Niederreiter encryption scheme describes codes through parity-check matrices. But both schemes have to hide any structure through a scrambling transformation and a permutation transformation. The encryption algorithm takes as input words of weight t where t is the number of errors that can be decoded. We denote by $\mathcal{W}_{q,n,t}$ the words of $\mathbb{F}_q^n$ of weight t. Figure 3 gives details of the encryption/decryption algorithms.

Fig. 3. Encryption and decryption algorithms of the Niederreiter cryptosystem

```
Encrypt(pk,$\boldsymbol{m} \in \mathcal{W}_{2,n,t}$)
1. Calculate $\boldsymbol{c} = \boldsymbol{H} \times \boldsymbol{m}^T$
2. Output $\boldsymbol{c}$
```

```
Decrypt(sk,$\boldsymbol{c} \in \mathbb{F}_2^{n-k}$)
1. Calculate $\boldsymbol{z} = \boldsymbol{S}^{-1} \times \boldsymbol{c}$
2. Calculate $\boldsymbol{y} = \gamma(\boldsymbol{z})$
3. Output $\boldsymbol{m} = \boldsymbol{y} \times \boldsymbol{P^{-1}}$
```

2.3 Stern identification and signature scheme

Stern's Scheme is an interactive zero-knowledge protocol which aims at enabling a *prover* P to identify himself to a *verifier* 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.

Each prover P receives a n-bit secret key sk (also denoted by s if there is no ambiguity about the prover) of Hamming weight t and computes a *public identifier* pk such that $\mathsf{pk} = \widetilde{H}\mathsf{sk}^T$.

When a user P needs to prove to V that he is indeed the person associated to the public identifier pk, then the two protagonists perform the protocol described in Figure 4.

The probability that a dishonest person succeeds in cheating is (2/3).

By virtue of the Fiat-Shamir Paradigm, it is possible to convert Stern's Protocol into a signature scheme.

Fig. 4. Stern identification scheme

1. P randomly chooses $y \in \mathbb{F}_2^n$ and a permutation σ of $\{1, 2, \ldots, n\}$. Then P sends to V the commitments c_1, c_2 and c_3 such that :
$$c_1 = h(\sigma \| \widetilde{H} y^T);\ c_2 = h(\sigma(y));\ c_3 = h(\sigma(y \oplus \mathsf{sk})),$$
where $h(a\|b)$ denotes the hash of the concatenation of the sequences a and b.
2. V sends $b \in \{0, 1, 2\}$ to P.
3. Three possibilities :
   - if $b = 0$: P reveals y and σ.
   - if $b = 1$: P reveals $(y \oplus \mathsf{sk})$ and σ.
   - if $b = 2$: P reveals $\sigma(y)$ and $\sigma(\mathsf{sk})$.
4. 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.

3 Previous works

In the literature, there exists only four articles dealing with side-channels attacks against code-based cryptosystems.

3.1 Side-channel attacks against the McEliece PKC

In their article [8], the authors have shown that the McEliece PKC like most known public key cryptosystems, bears a high risk of leaking secret information through side channels if the implementation does not feature appropriate countermeasures.

Furthermore, they presented a feasible power attack against the key generation phase, where certain operations involve the same secret value repeatedly. In general, key generation is a more difficult target for a side channel attack than decryption, because in contrast to that operation the attacker can only perform one measurement. But their considerations show, that without countermeasures, an implementation of the key generation might be vulnerable to a sophisticated power attack.

The cache attack designed to reveal the permutation that is part of the secret key, again benefits from the fact that the number of measurements the attacker may perform is in principle without any restraint. Thus the proposed secure algorithm seems to be an important countermeasure for software implementations intended for use in a multi user operating system.

Clearly, other parts of the cryptosystem require to be inspected with the same accuracy. This is especially true for the decryption phase, where the secret Goppa polynomial is employed in different operations.

Furthermore, [10] more closely inspects a timing side-channel already pointed out in [8]. This side-channel allows an attacker to find the plaintext to a given ciphertext given he has side-channel access to the decryption device holding the respective private key. The authors also devise an appropriate countermeasure that is implemented in the decoding algorithm.

In [11] a timing attack against the secret permutation and a corresponding countermeasure is presented.

3.2 SPA and first order DPA against Stern identification scheme

In [5], the authors described the first implementation of Stern protocol on smart card (in fact it is also more generally the first code-based system implemented on smart-card with usual resources). 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.

To secure the Stern scheme against SPA and first order DPA, the authors show that there are four parts of the protocol dealing with sensitive data :

- Matrix-vector Product
- Hash Function
- Permutation Method
- Pseudorandom Generator

They describe how to efficiently hide the leakage of information using random masks.

Moreover, the fact that the protocol essentially performs linear operations makes the algorithm easy to protect against side channel attacks.So, this protocol is a new option to carry out fast strong authentication on smart cards. Additionally, the use of a dedicated linear-algebra co-processor should significantly improve the timing performances of their implementation.

4 Conclusion

Code-based cryptosystems are a very attractive possibility in a post-quantum world. The operations involve in such schemes are very fast for encryption and decryption and there exists several interesting constructions based on hard problems of coding theory. The side-channel resistant implementation of such schemes has not been studied deeply so far and this article invite the *side-channel community* to study such possibilities. The McEliece PKC, though existing for 30 years, has not experienced wide use so far. But since it is one of the candidates for post quantum public key cryptosystems, it might become practically relevant in the near future. It is important to identify the potential side channels in a cryptosystem before it becomes commonly adopted.

Perspectives : In view of the small literature in this context, we encourage the community to look for side-channel attacks against the code-based cryptosystems like the Courtois-Finiasz-Sendrier signature scheme, the KKS signature scheme, the FSB hash function or the SYND stream cipher.

A complete list of code-based proposals can be found in [1].

References

1. http://cayrel.net/spip.php?article133
2. R. Misoczki and P. Barreto, *Compact McEliece Keys from Goppa Codes*, SAC'2009, http://eprint.iacr.org/2009/187
3. T. Berger, P.-L. Cayrel, P. Gaborit and A. Otmani, *Reducing Key Length of the McEliece Cryptosystem*, Africacrypt 2009, Lecture Notes in Computer Science, page 77–97
4. D.J. Bernstein, J. Buchmann and E. Dahmen, *Post-Quantum Cryptography*, Springer, Berlin, 2009, ISBN 978-3-540-88701-0.
5. P.-L. Cayrel, P. Gaborit and E. Prouff, *Secure Implementation of the Stern Authentication and Signature Schemes for Low-Resource Devices*, Eighth Smart Card Research and Advanced Application Conference CARDIS 2008 In G. Grimaud and F.-X. Standaert, editors, Lecture Notes in Computer Science, Vol. 5189, pages 191-205, 2008
6. Y. X. Li, R. H. Deng and X.-M. Wang, *On the equivalence of McEliece's and Niederreiter's public-key cryptosystems*, IEEE Transactions on Information Theory, volume 40, number 1, 1994, pages 271-273
7. R. J. McEliece, *A Public-Key System Based on Algebraic Coding Theory*, Jet Propulsion Lab, DSN Progress Report 44, 1978, pages 114-116
8. F. Strenzke,E. Tews, H. G. Molter, R. Overbeck and A. Shoufan *Side Channels in the McEliece PKC*, The Second international Workshop on Post-Quantum Cryptography PQCRYPTO 2008, Lecture Notes in Computer Science, Vol. 5299.
9. H. Niederreiter, *Knapsack-type cryptosystems and algebraic coding theory*, Problems Control Inform. Theory, Vol. 15, number 2, pages 159-166, 1986
10. A. Shoufan, F. Strenzke, H. G. Molter and M. Stöttinger *A Timing Attack Against Patterson Algorithm in the McEliece PKC*, in ICISC 2009
11. F. Strenzke *A Timing Attack against the secret Permutation in the McEliece PKC*, preprint 2010