Implementing CFS

Grégory Landais and Nicolas Sendrier

INRIA Paris-Rocquencourt, Project-Team SECRET
{gregory.landais,nicolas.sendrier}@inria.fr

Keywords: CFS, digital signature scheme, software implementation

CFS [1] is the first practical code-based signature scheme. It is based on the Niederreiter cryptosystem [2] and relies on the hardness of the syndrome decoding problem and on the undistinguishability of binary Goppa code.

There are relatively few instances of public-key digital signatures available and most of them are based on number theory. Even though, implementation issues related to CFS have received little attention. This can be explained by an (apparent) lack of practicality and also by some cryptanalytic results that have slightly weakened the scheme.

The purpose of this work is to (try to) clarify the situation, to show that the system is practical and secure when parameters are properly chosen.

We propose here a study of the implementation of the CFS (and the parallel-CFS countermeasure) schemes. Since it is the bottleneck of the implementation we will concentrate on the signing algorithm and thus on the decoding of Goppa codes. There exist alternatives for implementing finite field arithmetic (we need here extensions of the binary field of degree 16 to 24, which are not common) and for decoding (we will see that the usual choice, Patterson's algorithm, is not necessarily the best in this context). We explore these alternatives and extract guidelines for efficient implementation. To illustrate this study, we have designed a basic state-of-the-art software implementation of parallel-CFS (our target platform is a standard PC). In addition to timings, we have performed a precise field operation count which suggests that a faster field arithmetic may have a spectacular effect on the signing time. We also mean our algorithmic study to be the theoretical basis for dedicated coprocessors for signing with CFS.

We first review the CFS scheme, its attacks and the Parallel-CFS variant. This allows us to propose some sets of secure parameters. Next we describe the software implementation. More specifically, we explain the various algorithmic options for decoding binary Goppa and compare them. We conclude with some timings and detailed measurements of our implementation.

References

1. Courtois, N., Finiasz, M., Sendrier, N.: How to achieve a McEliece-based digital signature scheme. In Boyd, C., ed.: Advances in Cryptology - ASIACRYPT 2001. Volume 2248 of LNCS., Springer (2001) 157–174
2. McEliece, R.: A public-key cryptosystem based on algebraic coding theory. DSN Prog. Rep., Jet Prop. Lab., California Inst. Technol., Pasadena, CA (January 1978) 114–116