Fast Gröbner. Algebraic cryptanalysis of HFE and Filter Generators

Jean-Charles Faugère
SPACES /LIP6/Loria CNRS/Université Paris VI/INRIA
case 168, 4 pl. Jussieu, F-75252 Paris Cedex 05
E-mail: jcf@calfor.lip6.fr

HFE (Hidden Fields Equations) is a public key cryptosystem using polynomial operations over finite fields. It has been proposed by Jacques Patarin at Eurocrypt 96 following the ideas of Matsumoto and Imai. It has long been regarded as a very promising cryptosystem because it can be used to produce signatures as short as 128, 100 and even 80 bits. HFE gives the shortest (unbroken) signatures known except with the recent McEliece signature scheme. The idea of HFE is the following: a univariate polynomial $P(X)$ is chosen (the secret key). Next the univariate polynomial structure is hidden by replacing x by $\sum_{i=0} x_i w^i$ where w is a primitive root of $GF(2^n)$ and f_i is the coefficient of w^i in the previous expression. Now if P is of hamming weight 2 (resp. d) we obtain an algebraic system of degree 2 (resp. d) (it is the public key):

$$\{f_0 = f_1 = \cdots, f_{n-1} = 0\}$$

More precisely if $y \in GF(2^n)$ is the original message then $z_i = f_i(y)$ is the encrypted message. Recover the original message knowing the secret key is easy since it is equivalent to find the roots of a univariate polynomial; on the other hand with only the public key it is a difficult problem since it is equivalent to solve the *polynomial system*: $z_i = f_i(x_1, \ldots, x_n)$.

Hence to study HFE it is necessary to study and to solve polynomial system of equations. One of the most efficient tool for solving algebraic system is Gröbner bases (Buchberger). By computing a Gröbner basis of

$$V = \{(x_1, \ldots, x_n) \in GF(2)^n \mid f_1(x_1, \ldots, x_n) = \cdots = f_n(x_1, \ldots, x_n)\}$$

one can find the solutions of any system.

This is situation is very general in Cryptography (or even some decoding problem in Error Correcting Code): most of the cryptosystems can rewritten into algebraic equations; thus it is necessary to evaluate theoretically and practically the complexity of computing Gröbner bases over $GF(2)$. Note that this method (¡¡algebraic cryptanalysis¿¿) is completely automatic.

Another example of this reduction to algebraic equations is nonlinear filter generators. In such a device, a pseudo random sequence is generated as a non linear function f of the stages of a Linear Feedback Shift Register (LFSR). Thus, it is obvious that we can describe the generator by an algebraic system of N equations of degree d where N is the size of the output sequence and d the degree of the boolean function. It turns out, that if f is a k resilient boolean function with k small, then fast correlation attacks can be used; on the other side when k is greater than 2, the function f behaves like of a function of small degree and then Gröbner bases attacks can be used.

In this talk we present several new results:

- we describe briefly a new efficient algorithm for computing computing Gröbner bases over GF(2). This algorithm is several order of magnitude faster than any other algorithms. This is a fundamental tool for testing real size cryptosystems but also to derive useful theoretical bounds.

- by using this tool we can solve the *first HFE Challenge* of Patarin (corresponding to n=80,d=96) in only two days of CPU time. We are also able to find *experimentally* the complexity for solving HFE: for instance, when $d = 96$, HFE can be broken in $O(n^8)$ operations.

- We present some *theoretical* results concerning the complexity of solving a *random* system over $GF(2)$ (this is a work in common with B. Salvy and M. Bardet). We are able to predict then number and the size of all the matrices occuring in the algorithm. This can be used to *distinguish* a random system from a particular system of equations. For instance, the difference with a random system for the challenge 1 can be detected after 6 hours of computation.

- Another application of this theoretical study is the following result: computing a Gröbner basis of a (generic) algebraic system of $n\log(n)$ equations in n variables can be done in sub-exponential time. In other words, and transposed for filter generators it says: recovering the initial state of a non linear filter generator can be done in sub-exponential time when we have $n\log(n)$ bits of the output sequence. We present also simulation results on LFSR for standard benchmarks. The conclusion of these simulations, is that Gröbner bases can be used to attack *real size* (128 bits) LFSR (common work with G. Ars).

An open issue is to evaluate precisely the complexity of algebraic attacks for more difficult problem such as AES.