Practical Cryptanalysis of the Identification Scheme Based on the Isomorphism of Polynomial With One Secret Problem

Charles Bouillaguet, Jean-Charles Faugère, Pierre-Alain Fouque, Ludovic Perret

Practical Cryptanalysis of the Identification Scheme Based on the Isomorphism of Polynomial with One Secret Problem

Charles Bouillaguet[1], Jean-Charles Faugère[2,3],
Pierre-Alain Fouque[1] and Ludovic Perret[3,2]

[1] Ecole Normale Supérieure, Paris, France
{charles.bouillaguet, pierre-alain.fouque}@ens.fr
[2] SALSA Project - INRIA (Centre Paris-Rocquencourt)
UPMC, Univ Paris 06 - CNRS, UMR 7606, LIP6
104, avenue du Président Kennedy 75016 Paris, France
jean-charles.faugere@inria.fr, ludovic.perret@lip6.fr

Abstract. This paper presents a practical cryptanalysis of the Identification Scheme proposed by Patarin at Crypto 1996. This scheme relies on the hardness of the Isomorphism of Polynomial with One Secret (IP1S), and enjoys shorter key than many other schemes based on the hardness of a combinatorial problem (as opposed to number-theoretic problems). Patarin proposed concrete parameters that have not been broken faster than exhaustive search so far. On the theoretical side, IP1S has been shown to be harder than Graph Isomorphism, which makes it an interesting target. We present two new deterministic algorithms to attack the IP1S problem, and we rigorously analyze their complexity and success probability. We show that they can solve a (big) constant fraction of all the instances of degree two in polynomial time. We verified that our algorithms are very efficient in practice. All the parameters with degree two proposed by Patarin are now broken in a few seconds. The parameters with degree three can be broken in less than a CPU-month. The identification scheme is thus quite badly broken.

1 Introduction

Multivariate cryptography is concerned with the use of multivariate polynomials over finite fields to design cryptographic schemes. The use of polynomial systems in cryptography dates back to the mid eighties with the design of C* [32], and many others proposals appeared afterwards [36,37,38,27,46]. The security of multivariate schemes is in general related to the difficulty of solving random or structured systems of multivariate polynomial equations. This problem has been proved to be NP-complete [22], and it is conjectured [2] that systems of random polynomials are hard to solve in practice. As usual when a trapdoor must be embedded in a hard problem, easy instances are transformed into random-looking instances using secret transformations. In multivariate cryptography, it is common to map an easily-invertible collection of polynomials $\mathbf{a}$ into an apparently random one $\mathbf{b}$. It is then assumed that, being supposedly indistinguishable from random, $\mathbf{b}$ should be hard to solve. The structure-hiding transformation is very often the composition with linear (or affine) invertible mappings S and T, namely $\mathbf{b} = T \circ \mathbf{a} \circ S$. The matrices S and T are generally part of the secret-key.

The Isomorphism of Polynomials (IP) is the problem of recovering the secret transformations S and T given $\mathbf{a}$ and $\mathbf{b}$. It is a fundamental problem of multivariate cryptography, since its hardness implies the difficulty of the key-recovery for various multivariate cryptosystems. Notorious examples include C* [32], the traitor tracing scheme proposed by Billet and Gilbert [8], the SFLASH signature scheme [37], the ℓ-IC signature scheme [12], the square-vinegar signature scheme [1] and the Square encryption scheme [11][3]. All these schemes have been broken, because the structure of the central map was not hidden enough. The corresponding IP problem was then not random, but structured. However, when no apparent structure exists in both $\mathbf{a}$ and $\mathbf{b}$, then the IP problem is fairly difficult. This motivated Patarin to introduce it as an intractable assumption by itself in [34]. So far only exponential algorithms [39,17] are known to attack the general IP problem.

An important special case of IP is the *IP problem with one secret* (IP1S for short), where T is the identity matrix. Patarin suggested in 1996 [35] to construct a zero-knowledge identification scheme relying on the hardness of IP1S,

[3] In the description of some of these schemes, the easily-invertible central map contains parameters that are part of the secret-key. However, in this case there exists an equivalent secret key where these parameters have a fixed value. This is notoriously the case of C*.

inspired by the Zero-Knowledge proof system for Graph Isomorphism of [25]. The proposed parameters lead to relatively small key sizes (for instance to secret and public keys of 256 bits each and no additional information), as the complexity of the problem was believed to be exponential. The proposed parameters have not been broken so far, and no technique better than exhaustive search is known to attack the scheme. The IP1S problem is also interesting from a complexity-theoretic point of view. It has been proved in [39] that IP1S is *Graph Isomorphism-hard* (GI-hard for short). This leads Patarin *et al.* to claim that IP1S is unlikely to be solvable in polynomial time, because no polynomial algorithm is known for GI in spite of more than forty years of research. On the other hand, GI is not known to be NP-complete. Generating hard instances GI is pretty non-trivial, and there are powerful heuristics as well as expected linear time algorithms for random graphs [19]. This compromises the use of GI as an identification mechanism, and was part of the motivation for introducing IP1S as an alternative. Moreover, when used in this context, instances of the IP problem are random, which presumably avoids all the attacks on the cryptographic schemes mentioned above.

Previous and Related Work. The identification scheme based on IP1S is not based on number-theoretic assumptions, unlike for instance the well-known Fiat-Shamir protocol [18]. Many other identification schemes are not based on number theoretic assumptions [42,43,44,41,30]. However, the IP1S-based identification scheme enjoys shorter keys than most others.

To our knowledge, the first algorithm dedicated to IP1S can be found in Geiselmann *et al.* [23]. The authors of [23] remarked that each row of a matrix solution of IP1S verifies an algebraic system of equations. They then used an exhaustive search to find the solutions of such system. Soon after, this technique has been improved by Levy-dit-Vehel and Perret [13] who replaced this exhaustive search by a Gröbner basis computation. This still yields exponential algorithms, and the improvement induced by this replacement is is as significant as the gain obtained when comparing Gröbner basis and exhaustive search for solving random algebraic systems. It is negligible over small field (*i.e.*, typically, $\mathbb{F}_2$), but significant for instances of IP1S over large fields. However, the complexity of those algorithms remains exponential by nature.

Finally, Perret [40] shows that the affine and linear variants of IP1S are equivalent, *i.e.*, one can without loss of generality restrict our attention to the case where S is linear (as opposed to affine). In addition, a new approach for solving IP1S using the Jacobian matrix was proposed. The algorithm is polynomial when the number u of polynomials in $\mathbf{a}$ and $\mathbf{b}$ is equal to the number of variables n. However, when $u < n$, the complexity of this approach is not well understood. Moreover, when the number of polynomials is very small, for instance $u = 2$, this algorithm is totally inefficient.

The main application of IP1S is the identification scheme proposed in [39]. The public key being composed of two sets of u polynomials, it is interesting to keep the number of polynomials as small as possible (1 or 2). For such parameters, the authentication mechanism based on IP1S looks appealing in terms of key size.

All in all, the existing literature on the IP and IP1S problem can be split in two categories: *heuristic* algorithms with (more or less vaguely) "known" complexity and unknown success probability [39], and *rigorous* algorithms that always succeeds but with unknown complexity [17,40,13,23]. This situation makes it very difficult, if not plainly impossible to compare these algorithms based on their theoretical features. The class of instances that can be solved by a given algorithm of the first type is in general not known. Conversely, the class of instances over which an algorithm of the second type terminates quickly are often not known as well. This lead the authors of IP/IP1S algorithms to measure the efficiency of their techniques in practice, or even not to measure it at all. Several sets of concrete parameters for IP and IP1S were proposed by Patarin in [35], and can be used to measure the progress accomplished since their introduction. The techniques presented in this paper allow to break all these challenges in practice.

Techniques. The algorithms presented here are deterministic, and rely on the two weapons that have dealt a severe blow to multivariate cryptography: linear algebra and Gröbner bases. Our ideas borrow to the recent differential cryptanalysis of multivariate schemes. While the algorithms are not very complicated, analyzing their running time is fairly non-trivial, and requires the invocation of not-so-well-known results about linear algebra (such as the dimension of the commutant of a matrix, or the properties of the product of two skew-symmetric matrices), as well as known results about random matrices, most notably the distribution of the rank and the probability of being cyclic. The two most delicate steps of the analysis involve lower-bounding the dimension of the kernel of a homogeneous system of matrix equations, and upper-bounding the degree of polynomials manipulated by a Gröbner-basis algorithm.

Our Results. We present two new "rigorous" and deterministic algorithms. On the practical side, these algorithms are efficient: random quadratic IP1S instances and random cubic inhomogeneous IP1S instances can be broken in time $\mathcal{O}\left(n^6\right)$ for any size of the parameters. In particular, all the quadratic IP1S challenges proposed by Patarin are now broken in a few seconds. The biggest cubic IP1S challenge can be broken in less than 1 CPU-month. The IP1S identification scheme is thus broken beyond repair in the quadratic case. In the case of cubic IP1S, our attack runs in time $\mathcal{O}\left(n^6 \times q^n\right)$, and the security parameter have to be seriously reconsidered, which makes the scheme much less attractive, since the key size is cubic in n.

A rigorous analysis of our algorithms is both necessary and tricky. When generating linear equations, special care has to be taken to count how many of them are independent. The recent history of algebraic cryptanalysis taught us that failure to do so may have drastic consequences. Additionally, the complexity of Gröbner bases computation, even though a bit more well-understood now in the generic case, is still often a delicate matter for structured systems.

A unique and distinctive feature of our algorithms compared to the previous state of affairs, and one of our main theoretical contribution, is that we characterize the class of instances that can be solved by our techniques in polynomial time. We show, for instance, that a (big) constant fraction of all quadratic IP1S instances can be solved in polynomial time.

Organisation of the paper. In section 2, we recall some useful facts about the IP1S problem. Then, in section 3, we introduce the identification scheme based on the hardness of IP1S and compare it to other non-number theoretic based ID schemes. We then introduce our algorithms to break IP1S in the quadratic case in section 4, and in the cubic case in section 5.

2 The IP1S Problem

We recall the definition of the IP1S problem. Given two families of polynomials $\mathbf{a}$ and $\mathbf{b}$ in $\mathbb{F}_q[x_1, \ldots, x_n]^u$ the task is to find an invertible matrix $S \in \mathrm{GL}_n\left(\mathbb{F}_q\right)$ and a vector $c \in \left(\mathbb{F}_q\right)^n$ such that:

$$\mathbf{b}(\mathbf{x}) = \mathbf{a}(S \cdot \mathbf{x} + c). \tag{1}$$

We will denote by $f^{(k)}$ the homogeneous component of degree k of f, and by extension $\mathbf{a}^{(k)}$ denotes the vector of polynomials obtained by taking the homogeneous components of degree k of all the coordinates of $\mathbf{a}$. We define the derivative of $\mathbf{a}$ in c to be the function $\frac{\partial \mathbf{a}}{\partial c} : \mathbf{x} \mapsto \mathbf{a}(\mathbf{x} + c) + \mathbf{a}(\mathbf{x})$. The following lemma is very useful, and is at the heart of the techniques proposed in [17].

Lemma 1. *i) For all $k \geq 1$, we have:*

$$\mathbf{b}^{(k)} = \left(\mathbf{a} + \frac{\partial \mathbf{a}}{\partial c}\right)^{(k)} \circ S.$$

ii) If d is the degree of $\mathbf{a}$ and $\mathbf{b}$, then $\mathbf{b}^{(d)} = \mathbf{a}^{(d)} \circ S$.
iii) S transforms the set of common zeroes of $\mathbf{a}$ into the set of common zeroes of $\mathbf{b}$.

Proof. Let us write $T(\mathbf{x}) = T_\ell \cdot \mathbf{x} + T_c$ and $S(\mathbf{x}) = S_\ell \cdot \mathbf{x} + S_c$ where T_ℓ and S_ℓ are $n \times n$ matrices, whereas S_c and T_c are vectors of $\left(\mathbb{F}_q\right)^n$. We have:

$$\begin{aligned}
\mathbf{b}(\mathbf{x}) &= T_c + T_\ell \cdot \mathbf{a}(S_\ell \cdot \mathbf{x} + S_c) \\
&= T_c + T_\ell \cdot \Big(\mathrm{D}\mathbf{a}(S_\ell \cdot \mathbf{x}, S_c) + \mathbf{a}(S_\ell \cdot \mathbf{x}) + \mathbf{a}(S_c) - \mathbf{a}(0)\Big) \\
&= \Big[T_c + T_\ell \cdot \left(\mathbf{a}\left(S_c\right) - \mathbf{a}(0)\right)\Big] + \left(T_\ell \circ \frac{\partial \mathbf{a}}{\partial S_c} \circ S_\ell\right)(\mathbf{x}) + \left(T_\ell \circ \mathbf{a} \circ S_\ell\right)(\mathbf{x})
\end{aligned}$$

The first statement follows from the application of [17, lemma 4] to the last equality. The second and third statements are direct consequences of the first one. □

A useful consequence of lemma 1 is that without loss of generality we may assume c to be the null vector[4]. A consequence of point ii) is that from any instance of the problem we can deduce a *linear homogeneous* instance by considering only the homogeneous component of highest degree. If this instance can be solved, and S can be retrieved, then recovering c is not difficult, using a slight generalization of the idea shown in [24]. If S is known, then $\frac{\partial \mathbf{a}}{\partial c}$ can be explicitly computed, and c can usually be deduced therefrom. For instance, focusing on the homogeneous component of degree one yields a system of $u \cdot n$ linear equations in n variables that admits c as a solution. In most cases, it will in fact admit *only* c as a solution, which enables recovering c.

It was pointed out in [39] that if there is only one quadratic polynomial, then the problem is easily solved in polynomial time. This follows from the fact that quadratic forms admit a canonical representation (see for instance [29]). The change of coordinate can be then easily computed. We will therefore focus on the case of $u \geq 2$ when the polynomials are quadratic.

For various reasons, the IP1S problem becomes *easier* when u is close to n, and *harder* when u is small. For instance, the algorithm given in [40] deals with the case $u = n$ in polynomial time, but cannot tackle the case where $u = 2$ and n is big, which prevented it from breaking the parameters proposed by Patarin. Additionally, small values of u leads to smaller public keys. Therefore, we will restrict our attention to the case where $u = 2$ when the polynomials are quadratic, and where $u = 1$ when they are cubic. These are the most cryptographically relevant cases, and the most challenging. We will also consider the case where $\mathbb{F}_q$ is a field of characteristic two. It can be shown that this makes the problem a bit harder, but again this is the most cryptographically relevant case. The quadratic and cubic IP1S problems are very different and lead to specific approaches, therefore we will discuss them separately.

3 Patarin's IP1S-Based identification Scheme

Zero-Knowledge proofs were introduced in 1985 by Goldwasser, Micali and Rackoff in [26]. Soon afterwards, Fiat and Shamir [18] used the hardness of quadratic residuosity to build an efficient identification scheme. Many other identification schemes appeard afterwards, all relying on the hardness of number-theoretic assumptions. Some cryptographers took a different line of research, and tried to design identification scheme from different computational assumptions, not relying on number theory, but instead on the NP-hardness of some specific combinatorial problems.

One of the very-first combinatorial identification scheme was proposed by Shamir [42], and relied on the hardness of the *Permuted Kernel Problem* (PKP). Later on, Stern proposed in [43] a scheme based on the intractability of *Syndrome Decoding* (SD), and in [44] a scheme based on the intractability of *Constrained Linear Equations* (CLE). Finally, Pointcheval [41] proposed a scheme related to the hardness of the *Perceptron Problem*, originating from the area of learning theory. All these problems are NP-complete (as opposed to IP1S). The designers proposed practical parameters, aiming for a security level of 2^{64} or more, which are summarized in table 1. In all these schemes, it is required that all users share a public common set of information, a "common setting", usually describing the instance of the hard problem. For instance, in number-theoretic problems, the description of the curve, or of the group over which a discrete logarithm problem is considered is a common public information. While this information is not a "key" *stricto sensu*, it must nevertheless be stored by the prover and by the verifier, leading to higher memory requirements. However, in some case it can be chosen randomly, or generated online from a small seed using a PRNG.

Scheme	Common Setting	Public Key	Secret Key
PKP	2048	256	374
	7992	512	808
SD	131 072	256	512
	524 288	512	1024
CLE	3600	80	80
	3600	96	96
Perceptron	10807	144	117
IP1S	0	256	272

Table 1. Practical parameters proposed in [42,43,44,41,35] in order to obtain a security level of roughly 2^{64}.

[4] this was already observed in [40].

Challenge	n	q	Degree	Polynomial(s)	Public Key	Private Key
A	16	2	2	2	272 bits	256 bits
B	16	2	3	1	816 bits	256 bits
C	6	16	2	2	168 bits	144 bits
D	6	16	3	1	224 bits	144 bits
E	32	2	2	2	1056 bits	1024 bits

Table 2. Concrete parameters for IP1S. Patarin proposed challenges A,B,C and D in [35]. We introduce challenge E.

On the contrary, the IP1S-based identification scheme proposed by Patarin in [34,35] does not need the prover and the verifier to share additional information (except maybe the description of the finite field, which is very small). It works very similarly to the original identification scheme based on a zero-knowledge proof system for Graph-Isomorphism (GI) by Goldreich, Micali and Wigderson [25]. One of the reasons for replacing GI by IP1S is the existence of efficient heuristic algorithms for GI, capable of solving efficiently random instances. The generation of hard instances of GI is a delicate matter [19]. Replacing the GI problem by IP1S yields shorted key, and random instances of IP1S were *a priori* secure. Patarin proposed concrete parameters, which are shown in table. 2. The PKP and SD schemes lead to bigger keys than IP1S, while the Perceptron scheme leads to comparable key-sizes, and CLE yields smaller keys than IP1S, if we neglect the additional memory requirement imposed by the common setting.

These IP1S challenges cannot be attacked using the existing techniques [17,23,40]. So the best attack remains exhaustively searching for the secret key. As a final note, let us mention that Lyubashevsky recently proposed in [30] to build an identification scheme using the hardness of lattice problems, but did not propose concrete parameters.

4 Cryptanalysis of Quadratic IP1S

The main observation underlying our quadratic IP1S algorithm is that by *differentiating* equation (1), it is possible to collect linear equations between the coefficients of S and those of S^{-1}.

We denote by $\mathrm{D}f : (\mathbb{F}_q)^n \times (\mathbb{F}_q)^n \to {\mathbb{F}_q}^u$ the *differential* of a function $f : \mathbb{F}_q^n \to \mathbb{F}_q^u$. $\mathrm{D}f$ is defined by:

$$\mathrm{D}f(\mathbf{x}, \mathbf{y}) = f(\mathbf{x} + \mathbf{y}) - f(\mathbf{x}) - f(\mathbf{y}) + f(0)$$

It is easy to see that $\mathrm{D}f(\mathbf{x}, \mathbf{y}) = \mathrm{D}f(\mathbf{y}, \mathbf{x})$. If f is a polynomial of total degree d, then $\mathrm{D}f$ is a polynomial of total degree d, but of degree $d-1$ in $\mathbf{x}$ and $\mathbf{y}$. Thus, when f is quadratic, then $\mathrm{D}f$ is a symmetric *bilinear* mapping.

Going back to the quadratic IP1S problem, for all vectors $\mathbf{x}, \mathbf{y} \in (\mathbb{F}_q)^n$, we have:

$$\forall \mathbf{x}, \mathbf{y} \in (\mathbb{F}_q)^n, \quad \mathrm{D}\mathbf{b}(\mathbf{x}, \mathbf{y}) = \mathrm{D}\mathbf{a}\left(S \cdot \mathbf{x}, S \cdot \mathbf{y}\right).$$

Using the change of variable $\mathbf{y}' = S \cdot \mathbf{y}$, this equation becomes:

$$\forall \mathbf{x}, \mathbf{y}' \in (\mathbb{F}_q)^n, \quad \mathrm{D}\mathbf{b}(\mathbf{x}, S^{-1} \cdot \mathbf{y}') = \mathrm{D}\mathbf{a}(S \cdot \mathbf{x}, \mathbf{y}'). \tag{2}$$

Since $\mathbf{a}$ and $\mathbf{b}$ are of total degree 2, then $\mathrm{D}\mathbf{a}$ and $\mathrm{D}\mathbf{b}$ are *bilinear* (symmetric) mappings. In this case, since equation (2) is valid for all $\mathbf{x}$ and $\mathbf{y}$, then in particular it is valid on a basis of $(\mathbb{F}_q)^n \times (\mathbb{F}_q)^n$, and substituting fixed basis vectors for $\mathbf{x}$ and $\mathbf{y}$ yields *linear equations* between the coefficients of S and those of S^{-1}.

This idea for obtaining linear equations can also be described relatively simply using the usual theory of quadratic forms (with the tweaks required by the fact that we are working in characteristic two). If $\mathbb{F}_q$ is a field of even characteristic, then the set of homogeneous quadratic polynomials in n variables over $\mathbb{F}_q$ is in one-to-one correspondance with the set of symmetric matrices with zero diagonal. Let $\mathcal{P}(\mathbf{a}_k)$ denote the matrix of the symmetric bilinear form associated with $\mathbf{a}_k$ (it is related to the *polar form* of $\mathbf{a}_k$ in odd characteristic). Recall that the coefficient of index (i, j) of $\mathcal{P}(\mathbf{a}_k)$ is $\mathrm{D}\mathbf{a}_k(e_i, e_j)$, where $(e_i)_{1 \le i \le n}$ is a basis of $(\mathbb{F}_q)^n$. We then have:

$$\mathcal{S}: \quad \begin{cases} S^{-1} \cdot \mathcal{P}(\mathbf{b}_1) = \mathcal{P}(\mathbf{a}_1) \cdot {}^tS \\ \vdots \\ S^{-1} \cdot \mathcal{P}(\mathbf{b}_u) = \mathcal{P}(\mathbf{a}_u) \cdot {}^tS \end{cases} \tag{3}$$

Each one of these u matrix equations yields n^2 linear homogeneous equations between the $2n^2$ coefficients of S and those of S^{-1}. These last $u \cdot n^2$ homogeneous linear equations cannot be linearly independent as they admit a non-trivial solution $\left(S^{-1}, S\right)$. The kernel of $\mathcal{S}$ is thus non-trivial, and our hope would be that it describes *only* one solution. When u is strictly greater than two, we then have much more linear equations than unknowns, and we empirically find only one solution (when the polynomials are randomly chosen). When $u = 2$, which is again the most relevant case, the situation is unfortunately not as nice; Theorem 1 below shows that the kernel of $\mathcal{S}$ is of dimension higher that n – in fact, we will show below that it is of dimension at least $2n$. This means that solving the linear equations cannot by itself reveal the solution of the IP1S problem, because $\mathcal{S}$ admits at least q^{2n} solutions, out of which only very few are actual solutions of the IP1S instance[5]. However, the linear equations collected this way can be used to simplify the resolution of the IP1S problem.

When looking at one coordinate of (1), we have an equality between two multivariate polynomials that holds for any value of the variables. Therefore the coefficients of the two polynomials can be identified (this is essentially the algorithm presented in [17]). This yields a system $\mathcal{S}_{\mathrm{quad}}$ of $u \cdot n^2/2$ quadratic equations in n^2 unknown over $\mathbb{F}_q$. With $u = 2$, this precisely gives n^2 equations in n^2 unknown, which cannot be solved by any existing techniques faster than exhaustive search.

However, we now know that S lives in the kernel of $\mathcal{S}$, and therefore S can be written as the sum of $k = \dim \ker \mathcal{S}$ matrices that can be easily computed using standard linear algebra. Identifying coefficients in (1) then yields a system $\mathcal{S}_{\mathrm{quad}}$ of $u \cdot n^2/2$ quadratic equations in k unknown. Our hope is that k is small enough for the system to be very overdefined, so that computing a Gröbner basis of $\mathcal{S}_{\mathrm{quad}}$ is polynomial in theory, and feasible in practice.

The analysis of the attack then proceeds in two steps:

1. Estimate the rank of $\mathcal{S}$ (*i.e.*, the value of k).
2. Estimate the complexity of the Gröbner basis computation.

For the sake of simplicity, we will analyze the attack algorithm under some assumptions on the input system. For instance, we will assume that that n is even, and that one of the two quadratic forms we are dealing with is non-degenerate. We will then argue that a random instance satisfies this assumption with high probability, but we are well aware that some structured instance may not. This is in fact quite logical, because a worst-case polynomial algorithm for IP1S would imply a worst-case polynomial for Graph-Isomorphism (a fact that would be quite surprising). The situation of the IP1S problem is in this respect quite similar to that of GI: heuristics are capable of dealing efficiently with the random case, while some very special instances make them fail (interestingly, hard instances for GI are transformed into hard instances for IP1S through the reduction). Lastly, we mention that our algorithm does not necessarily fail on an instance that does not meet our assumptions. However, we no longer have a guarantee on its running time. Random instances fail to meet the assumption with a small probability, but we empirically observed that the algorithm solves them in reasonable time as well.

4.1 Counting Linearly Independent Equations

Obtaining guarantees on the number of linearly independent equations in $\mathcal{S}$ is the most important and the most delicate part of the attack. Since $\dim \ker \mathcal{S}$ is a function of the instance, it makes sense to consider the random variable giving $\dim \ker \mathcal{S}$ assuming the instance was randomly chosen. Fig. 1 above shows its (experimentally observed) distribution for various sizes of the base field. We immediately see that in odd characteristic, $\dim \ker \mathcal{S}$ is often n, while in characteristic two it is often $2n$. In the sequel we provide mathematical arguments to back this observation up. We will focus on the (harder) case of fields of characteristic two, since this is the more cryptographically relevant case.

Our results are expressed in terms of the *similarity invariants* $P_1, \ldots, P_s$ of a matrix M. Their product is the characteristic polynomial of M, P_s is the minimal polynomial of M, and P_i divides P_{i+1}. The main technical result needed to understand the rank of $\mathcal{S}$ is the following theorem.

Theorem 1. *Let A_1, A_2, B_1, B_2 be four given matrices of size $n \times n$ with coefficients in $\mathbb{F}_q$. Let us consider the set of all pairs (X, Y) of $n \times n$ matrices satisfying the following linear equations:*

$$\mathcal{S}: \quad \begin{cases} B_1 = X \cdot A_1 \cdot Y \\ B_2 = X \cdot A_2 \cdot Y \end{cases}$$

Let us assume that $\mathcal{S}$ admits at least one solution (X_0, Y_0) with both X_0 and Y_0 invertible, and that A_1 is also invertible.

[5] We note that this contradicts the hope expressed in section 9 of [39]

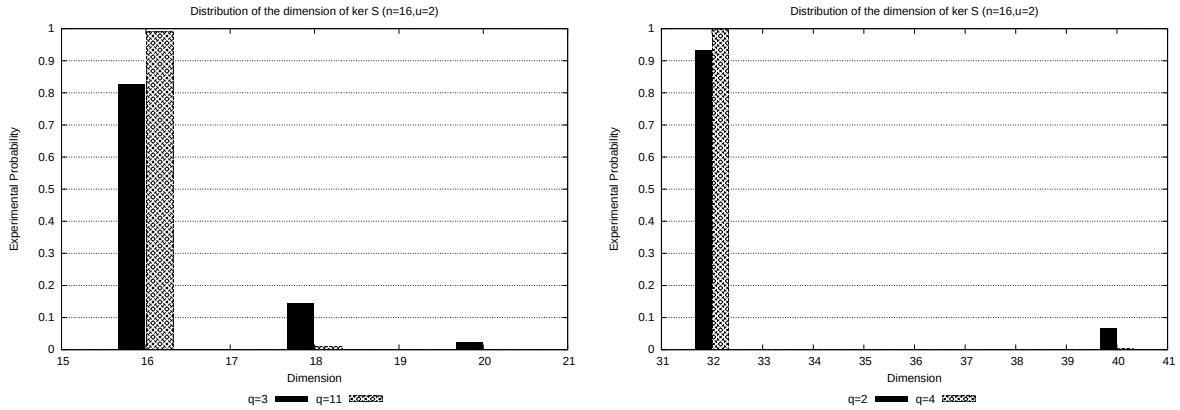

Fig. 1: Experimental distribution of $\dim \ker \mathcal{S}$.

i) There is a vector-space isomorphism between the kernel of $\mathcal{S}$ and the commutant of $\mathcal{C} = A_2 \cdot A_1^{-1}$.
ii) $n \leq \dim \ker \mathcal{S}$.
iii) Let $P_1, \ldots, P_s$ be the similarity invariants of $\mathcal{C}$. Then:

$$\dim \ker \mathcal{S} = \sum_{j=1}^{s} (2s - 2j + 1) \cdot \deg P_j$$

Proof. Because a solution of $\mathcal{S}$ exists, then B_1 is invertible. Thanks to this, we can write:

$$\mathcal{S}: \quad \begin{cases} Y = A_1^{-1} \cdot X^{-1} \cdot B_1 \\ B_2 \cdot B_1^{-1} \cdot X = X \cdot A_2 \cdot A_1^{-1} \end{cases}$$

Using the particular solution X_0 then gives:

$$\mathcal{S}: \quad \begin{cases} Y = A_1^{-1} \cdot X^{-1} \cdot B_1 \\ \mathcal{C} \cdot \left(X_0^{-1} \cdot X\right) = \left(X_0^{-1} \cdot X\right) \cdot \mathcal{C} \end{cases}$$

From there, it is not difficult to see that the kernel of $\mathcal{S}$ is in one-to-one correspondance with the commutant of $\mathcal{C}$, the isomorphism being $(X, Y) \mapsto X_0^{-1} \cdot X$. The second point of the theorem follows from the well-known fact that n lower-bounds the dimension of the commutant of any endomorphism on a vector space of dimension n (see for instance [7, Fact 2.18.9]). The third point follows from a general result on the dimension of the commutant [20, chapter 6, exercise 32]. □

Theorem 1 directly applies to our study of the rank of $\mathcal{S}$ with $A_i = \mathcal{P}(\mathbf{a}_i)$ and $B_i = \mathcal{P}(\mathbf{b}_i)$. However, it holds only if $\mathcal{P}(\mathbf{a}_1)$ or $\mathcal{P}(\mathbf{a}_2)$ is invertible (we may swap them if we wish, or even take a linear combination). Note that since $\mathcal{P}(\mathbf{a}_1)$ is a random skew-symmetric matrix, it cannot be invertible if n is odd, and the analysis is more complicated in that case. This is why we focus on the case where n is even, and where one of the two quadratic forms is non-degenerate. The following lemma gives us the probability that $\mathcal{P}(\mathbf{a}_i)$ (or $\mathcal{P}(\mathbf{b}_i)$) is invertible.

Lemma 2 ([31], theorem 3). *Let $N_0(n, r)$ denote the number of symmetric matrices of size $n \times n$ over $\mathbb{F}_q$ with zeros on the diagonal and of rank r.*

$$N_0(n, 2s) = \prod_{i=1}^{s} \frac{q^{2i-2}}{q^{2i} - 1} \cdot \prod_{i=1}^{2s-1} \left(q^{n-i} - 1\right)$$
$$N_0(n, 2s+1) = 0$$

If n is even, the probability that $\mathcal{P}(\mathbf{a}_1)$ is invertible if $q = 2$ is about 0.419 (this probability increases exponentially with q). The probability that either $\mathcal{P}(\mathbf{a}_1)$ or $\mathcal{P}(\mathbf{a}_2)$ is invertible is then about 0.662 when $q = 2$.

Theorem 1 is then applicable in more than half of the cases when $q = 2$ (and we expect this proportion to grow very quickly with q). When it is applicable, what guarantee does it exactly offer? We would need to know something about the similarity invariants of $\mathcal{C}$. An easy case would be when the minimal and characteristic polynomials are the same (then there is only one invariant factor, and it is precisely the characteristic polynomial). Then Theorem 1 tells us that the dimension of $\ker \mathcal{S}$ is n. For random matrices, the probability of this event is given by the following lemma.

Lemma 3 ([21], theorem 1). *Let $c(n, q)$ be the proportion of cyclic $n \times n$ matrices (i.e., matrices for which the minimal polynomial is of degree n). We have:*

$$\frac{1}{q^2(q+1)} < 1 - c(n,q) < \frac{1}{(q^2-1)(q-1)}$$

And asymptotically, we have:

$$\lim_{n\to\infty} c(n,q) = \frac{q^5-1}{q^2(q-1)(q^2-1)} \cdot \prod_{i=1}^{\infty}\left(1 - \frac{1}{q^i}\right)$$

For random matrices over $\mathbb{F}_2$, and for n big enough, the proportion of cyclic matrices approaches 0.746. Unfortunately, $\mathcal{C}$ is hardly a random matrix. In odd characteristic it is the product of two symmetric matrices, while in characteristic two it is the product of two symmetric matrices with null diagonal. The product of two such matrices is *very far* from being random, and it is in fact *never* cyclic, as the following result shows.

Theorem 2 ([6]). *Let M be a non-singular matrix of even dimension. Then the two following conditions are equivalent:*

i) M can be written as the product of two symmetric matrices with null-diagonal.
ii) M has an even number of similarity invariants $P_1, \ldots P_{2\ell}$, and $P_{2i+1} = P_{2i+2}$.

Corollary 1. *If n is even and $\mathcal{C}$ is invertible, then $\ker \mathcal{S}$ has dimension at least $2n$.*

Proof. By theorem 2, $\mathcal{C}$ has at least two invariants, both equal to the minimal polynomial of $\mathcal{C}$ (which thus happens to be of degree $n/2$). Then theorem 1, point iii) shows that $\ker \mathcal{S}$ has dimension $2n$. If $\mathcal{C}$ has more invariants, $\ker \mathcal{S}$ can only be of higher dimension. □

Corollary 1 shows that with a constant probability (when the two quadratic forms are non-degenerate) $\dim \ker \mathcal{S}$ is greater than $2n$, which sounds like bad news. When $\mathcal{C}$ is not invertible, theorem 2 no longer holds (there are counter-examples), but what does apparently still hold is the fact that the minimal polynomial of $\mathcal{C}$ has degree at most $n/2$, and this would be sufficient to show that in all cases $\dim \ker \mathcal{S} \geq 2n$, in accordance with Fig. 1.

What we would in fact need to know is the probability that $\ker \mathcal{S}$ is exactly of dimension $2n$. Theorem 1 still connects this dimension to the similarity invariants of $\mathcal{C}$, even though $\mathcal{C}$ is not a uniformly random matrix. It seems plausible that $\mathcal{C}$ is unlikely to have a very high number of similarity invariants, and that the most common situation is that it has only two invariants (twice the minimal polynomial). We could not compute explicitly this probability, and we could not find ways to obtain it in the available literature. We measured it experimentally and found 0.746 (after 10^5 trials) when $q = 2$. This is strikingly close to the result brought by lemma 3 in the random case. Under the conjecture that $\mathcal{C}$ has two invariant factors with this probability, then theorem 1 tells us that in about 75% of the cases, $\dim \ker \mathcal{S} = 2n$. The empirical probability seems to be even higher, as shown by Fig 1.

4.2 Solving Very Overdefined Quadratic Systems

The solution of the IP1S instance (1) is systematically the solution of a system $\mathcal{S}_{\text{quad}}$ of n^2 quadratic equations. In the previous section, we argued that we can reduce this system to n^2 equations in $2n$ unknowns with high probability, and (much) more unknowns with negligible probability. The system is so overdefined that it can almost be resolved by linearization. Indeed, it has $N^2/4$ equations in N unknowns. In practice, computing a Gröbner basis of the ideal generated by $\mathcal{S}_{\text{quad}}$ terminates very quickly, and allows to recover the actual solutions of the problem.

This last fact can be theoretically justified. It is well-known that Gröbner basis algorithms [15,16] are more efficient on overdefined systems. The complexity of most algorithms strongly depend on a parameter of the ideal called the *degree of regularity*. Indeed, the cost of computing a Gröbner basis is polynomial in the degree of regularity D_{reg} of the system when the ideal has dimension zero, *i.e.*, when the number of solutions is finite. The computation of a Gröbner basis essentially amounts to solve a system of M sparse linear equations in M variables, where M is the number of monomials of degree D_{reg} in N variables. The complexity of this process is roughly $\mathcal{O}\left(N^{3\cdot D_{\mathrm{reg}}}\right)$, with $2 < \omega \leq 3$ the linear algebra constant, and N the number of variables of ideal considered (in our case, $N = 2n$).

The behavior of the degree of regularity is well understood for "random" systems of equations [3,4,5] (*i.e., regular* or *semi-regular* systems). It is conjectured that the proportion of semi-regular systems on N variables goes to 1 when N goes to $+\infty$. Therefore, we can assume that for large N a random system is almost surely semi-regular. This is to some extent a worst-case assumption, as it usually means that our system is not easier to solve than the others. The coefficients of the Hilbert series associated with the ideal generated by a semi-regular sequence of m equations in N variables coincide with those of the series expansion of the function $f(z) = \left(1 - z^2\right)^m / (1-z)^N$, up to the degree of regularity. The degree of regularity is the smallest degree d such that the coefficient of degree d in the series expansion of $f(z)$ is not strictly positive. This property enables an explicit computation of the degree of regularity for given values of m and N.

Furthermore, the available literature readily provide asymptotic estimates of the degree of regularity for semi-generic ideals of $N + k$ or $\alpha \cdot N$ equations in N variables, but unfortunately not for the case of $\alpha \cdot N^2$ in N variables, which is the situation we are facing here. We thus tabulated in table. 3 the degree of regularity for semi-regular systems of equations having the same number of equations and unknowns as those occurring in our attack. From this table, we conclude that for any reasonable value of the parameters, the degree of regularity will be 3, and thus computing a Gröbner basis of $\mathcal{S}_{\mathrm{quad}}$ should have complexity at most $\mathcal{O}\left(n^9\right)$. In practice, the maximal degree reached by the F_4 algorithm on our equations is two, which is even better.

n	2	3	4	5	6	7	8	...	16	...	32
N	4	6	8	10	12	14	16	...	32	...	64
m	4	9	16	25	36	49	64	...	256	...	1024
D_{reg}	5	4	3	3	3	3	3	...	3	...	3

Table 3. Degree of regularity of random with the same parameters as those occuring in our attack.

4.3 Implementation

We demonstrated that the algorithm described in this section terminates in time $\mathcal{O}\left(n^6\right)$ on a constant fraction of the instances. This reasoning is backed up by empirical evidence: we implemented the algorithm using the computer algebra system MAGMA [9]. Solving the equations of $\mathcal{S}_{\mathrm{quad}}$ is achieved by first computing a Gröbner basis of these equations for the Graded-Reverse Lexicographic order using the F_4 algorithm [15], and then converting it to the Lexicographic order using the FGLM algorithm [14]. This implementation breaks the random instances of IP1S in very practical time. For instance, Challenges A and C are solved in a few seconds. Challenge E takes a few minutes, but the dominating part in the execution of the algorithm is in fact the symbolic manipulation of polynomials required to write down the equations of $\mathcal{S}_{\mathrm{quad}}$. Actually solving the resulting quadratic equations turns out to be easier than generating them. We never generated a random instance that we could not solve with our technique, for any choice of the parameters.

There are only public parameter sets, and no public challenges to break, so we unfortunately cannot provide the solution of an open challenge to prove that our algorithm works. However, the source code of our implementation is available on the webpage of the first author.

5 Cryptanalysis of Cubic IP1S

In this section, we focus on the case where $\mathbf{a}$ and $\mathbf{b}$ are composed of a single cubic polynomial. We assume that $\mathbf{a}$ and $\mathbf{b}$ are given explicitly, i.e.:

$$\mathbf{a} = \sum_{i=1}^{n}\sum_{j=i}^{n}\sum_{k=j}^{n} A_{i,j,k} \cdot x_i x_j x_k, \qquad \mathbf{b} = \sum_{i=1}^{n}\sum_{j=i}^{n}\sum_{k=j}^{n} B_{i,j,k} \cdot x_i x_j x_k.$$

As already explained, we can restrict our attention to the homogenous case. The techniques developed previously for the quadratic case cannot directly applied in this setting. Indeed, the differential is no longer a bilinear mapping, and then there is no obvious linear equations between the coefficients of a solution and those of its inverse. However, we can combine the use of the differential together with the Gröbner basis approach proposed in [17]. We denote by $S_0 = \{s^0_{i,j}\}_{1\leq i,j\leq n}$ a particular solution of IP1S between $\mathbf{a}$ and $\mathbf{b}$, *i.e.*, it holds that $\mathbf{b} = \mathbf{a} \circ S_0$. For all vectors $\mathbf{x}, \mathbf{y} \in (\mathbb{F}_q)^n$, we have:

$$\mathrm{D}\mathbf{a}(S_0 \cdot \mathbf{x}, \mathbf{y}) = \mathrm{D}\mathbf{b}(\mathbf{x}, S_0^{-1} \cdot \mathbf{y}).$$

$\mathbf{a}$ and $\mathbf{b}$ being of total degree 3, the coefficients of S_0 and S_0^{-1} appear with degree two in the expression of $\mathrm{D}\mathbf{a}$ and $\mathrm{D}\mathbf{b}$ above. Let R be the ring $\mathbb{K}[s_{1,1}, \ldots, s_{n,n}, u_{1,1}, \ldots, u_{n,n}]$. We consider the algebra $\mathcal{A}^s$ of all $n \times n$ matrices over R. Let $S = \{s_{i,j}\}$ and $U = \{u_{i,j}\}$ in $\mathcal{A}^s$ be symbolic matrices. We denote by $\mathcal{I}_{\mathbf{a},\mathbf{b}}$ the ideal generated by all the coefficients in R of the equations:

$$\mathrm{D}\mathbf{a}(S \cdot \mathbf{x}, \mathbf{y}) - \mathrm{D}\mathbf{b}(\mathbf{x}, U \cdot \mathbf{y}) = 0, \qquad U \cdot S - 1_n = 0_n, \qquad S \cdot U - 1_n = 0_n.$$

It is easy to see that $U = S_0^{-1}$ and $S = S_0$ is particular solution of this system, and also a solution of IP1S between $\mathbf{b}$ and $\mathbf{a}$. Our goal is to provide an upper bound on the maximum degree reached during a Gröbner basis computation of $\mathcal{I}_{\mathbf{a},\mathbf{b}}$.

We prove here that $D_{\mathrm{reg}} = 2$ for $\mathcal{I}_{\mathbf{a},\mathbf{b}}$ under the hypothesis that we know one row of a particular solution S_0, *i.e.*, we assume then that we know the following ideal $\mathcal{J} = \left\langle s_{1,j} - s^{(0)}_{1,j} \mid j = 1, \ldots, n \right\rangle$.

Theorem 3. *The degree of regularity of $\mathcal{I}_{\mathbf{a},\mathbf{b}} + \mathcal{J}$ is 2. Therefore, computing a Gröbner basis of this ideal takes time $\mathcal{O}\left(n^6\right)$.*

Proof. We use the fact that the degree of regularity of an ideal is generically left invariant by any linear change of the variables or generators [28]. In particular, we consider the ideal $\mathcal{I}'_{\mathbf{a},\mathbf{b}}$ generated by all the coefficients in $\mathbb{K}[x_1, \ldots, x_n, y_1, \ldots, y_n]$ of the equations:

$$\mathrm{D}\mathbf{a}(S_0(S + I_n)\mathbf{x}, \mathbf{y}) - \mathrm{D}\mathbf{b}(\mathbf{x}, (U + I_n)S_0^{-1}\mathbf{y}) = 0, \qquad U \cdot S = 0_n, \qquad S \cdot U = 0_n.$$

It is clear that $\mathcal{I}'_{\mathbf{a},\mathbf{b}}$ is obtained from $\mathcal{I}_{\mathbf{a},\mathbf{b}}$ by replacing S (resp. U) by $S_0(I_n + S)$ (resp. $(U + I_n)S_0^{-1}$). Thus, the degree of regularity of $\mathcal{I}'_{\mathbf{a},\mathbf{b}}$ and $\mathcal{I}_{\mathbf{a},\mathbf{b}}$ are equal. Using the same transformation, the ideal $\mathcal{J}$ becomes

$$\mathcal{J}' = \langle s_{1,j} \mid j = 1, \ldots, n \rangle .$$

We now estimate the degree of regularity of the ideal $\mathcal{I}'_{\mathbf{a},\mathbf{b}} + \mathcal{J}'$. For a reason which will become clear in the sequel, it is more convenient to work with $\mathcal{I}'_{\mathbf{a},\mathbf{b}} + \mathcal{J}'$. In what follows, F will denote the generators of $\mathcal{I}'_{\mathbf{a},\mathbf{b}} + \mathcal{J}'$. We will show that many new linear equations appear when considering equations of degree 2. To formalize this, we introduce some definitions related to the F_4 algorithm [16]. In particular, we will denote by $I_{d,k}$ the linear space generated during the k-th step of F_4 when considering polynomials of degree d.

Definition 1. *We have the following recursive definition of $I_{d,k}$:*

$$\begin{aligned}
I_{d,0}(F) &= \mathit{Vect}_{\mathbb{K}}\left(F\right)\\
I_{d,1}(F) &= \mathit{Vect}_{\mathbb{K}}\left(s_{i,j}f \mid 1 \leqslant i,j \leqslant n \text{ and } f \in I_{d,0}(F)\right)\\
&\quad + \mathit{Vect}_{\mathbb{K}}\left(u_{i,j}f \mid 1 \leqslant i,j \leqslant n \text{ and } f \in I_{d,0}(F)\right)\\
I_{d,k}(F) &= \mathit{Vect}_{\mathbb{K}}\left(s_{i,j}f \mid 1 \leqslant i,j \leqslant n \text{ and } f \in I_{d,k-1}(F) \text{ and } \deg(f) \leq d-1\right)\\
&\quad + \mathit{Vect}_{\mathbb{K}}\left(u_{i,j}f \mid 1 \leqslant i,j \leqslant n \text{ and } f \in I_{d,k-1}(F) \text{ and } \deg(f) \leq d-1\right).
\end{aligned}$$

Roughly speaking, the index k is the number of steps in the F_4/F_5 [16] algorithm to compute an element $f \in I_{d,k}(F)$. We show that $I_{2,1}(F)$ contains exactly n^2+2n linear equations. This means that we have already many linear equations generated during the first step of a Gröbner basis computation of F.

Lemma 4. *$I_{2,1}(F)$ contains the following linear equations:*

$$\{u_{1,j} \mid j = 1, \ldots, n\}. \tag{4}$$

Proof. From the first row of the following zero matrix $S \cdot U$ we obtain the following equations:

$$\begin{cases} s_{1,1}\,u_{1,1} + s_{1,2}\,u_{2,1} + s_{1,3}\,u_{3,1} + \cdots + s_{1,n}\,u_{n,1} = 0, \\ s_{1,1}\,u_{1,2} + s_{1,2}\,u_{2,2} + s_{1,3}\,u_{3,2} + \cdots + s_{1,n}\,u_{n,2} = 0, \\ s_{1,1}\,u_{1,3} + s_{1,2}\,u_{2,3} + s_{1,3}\,u_{3,3} + \cdots + s_{1,n}\,u_{n,3} = 0, \\ \quad \cdots \\ s_{1,1}\,u_{1,n} + s_{1,2}\,u_{2,n} + s_{1,3}\,u_{3,n} + \cdots + s_{1,n}\,u_{n,n} = 0 \end{cases}$$

Using the equations $s_{1,j} = 0$ from the ideal $\mathcal{J}'$, we obtain then $u_{1,1} = 0, u_{1,2} = 0, \ldots, u_{1,n} = 0$. □

We can also predict the existence of other linear equations in $I_{2,1}(F)$.

Lemma 5. *For all $(i,j) \in \{1,\ldots,n\}^2$ the coefficient of $y_1 y_i x_j$ in $\mathrm{D}\mathbf{a}(S_0(S+I_n)\mathbf{x},\mathbf{y}) - \mathrm{D}\mathbf{b}(\mathbf{x},(U+I_n)S_0^{-1}\mathbf{y})$ is a non zero[6] linear equation modulo the equations of the ideal $\mathcal{J}'$ and* (4)*. Among these equations, there are n which depend only of the variables $\{s_{k,\ell} \mid 1 \le k,\ell \le n\}$.*

Proof. We consider the coefficient of the monomial $m = y_1 y_i x_j$ in the expression

$$\Delta = \Delta_a - \Delta_b = \mathrm{D}\mathbf{a}(S_0(S+I_n)\mathbf{x},\mathbf{y}) - \mathrm{D}\mathbf{b}(\mathbf{x},(U+I_n)S_0^{-1}\mathbf{y}).$$

Since the monomial m is linear in x_j it is clear that the corresponding coefficient in $\Delta_a = \mathrm{D}\mathbf{a}(S_0(S+I_n)\mathbf{x},\mathbf{y})$ is also linear in the variables $s_{i,j}$; moreover this coefficient is non zero. We have now to consider the coefficient of m in Δ_b. Since $\mathrm{D}\mathbf{b}(\mathbf{x},\mathbf{y})$ is the differential of an homogenous polynomial of degree 3 we can always write:

$$\mathrm{D}\mathbf{b}(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n}\sum_{j=i}^{n} \ell_{i,j}(y_1,\ldots,y_n)\,x_i x_j + \sum_{i=1}^{n} q_i(y_1,\ldots,y_n)x_i \tag{5}$$

where $\ell_{i,j}$ (resp. q_i) is a polynomial of degree 1 (resp. 2). Consequently, the coefficient of m in $\mathrm{D}\mathbf{b}$ is also the coefficient of $y_1 y_i$ in $q_j((U+I_n)S_0^{-1}\mathbf{y})$. That is to say, in $q_j(\mathbf{y})$ we have now to replace $\mathbf{y} = (y_1,\ldots,y_n)$ by $((U+I_n)S_0^{-1}\mathbf{y})$. Thus, modulo the equations of the ideal $\mathcal{J}'$ and (4), we can write the product $((U+I_n)S_0^{-1}\mathbf{y})$ as

$$= \begin{pmatrix} y_1 \\ \vdots \\ \vdots \\ y_n \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ u_{2,1} & \cdots\cdots & & u_{2,n} \\ \vdots & \cdots\cdots & & \vdots \\ u_{n,1} & \cdots\cdots & & u_{n,n} \end{pmatrix} \begin{pmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{pmatrix}$$

$$= \begin{pmatrix} y_1 \\ \vdots \\ \vdots \\ y_n \end{pmatrix} \begin{pmatrix} * & & * & * & & * \\ (*u_{2,1} + \cdots + *u_{2,n}) & \cdots\cdots & & & \cdots\cdots & (*u_{2,1} + \cdots + *u_{2,n}) \\ \vdots & & \cdots\cdots & & & \vdots \\ (*u_{n,1} + \cdots + *u_{n,n}) & \cdots\cdots & & & \cdots\cdots & (*u_{2,1} + \cdots + *u_{n,n}) \end{pmatrix}$$

$$= \begin{pmatrix} *y_1 + (*u_{2,1} + \cdots + *u_{2,n})y_2 + \cdots + (*u_{n,1} + \cdots + *u_{n,n})y_n \\ *y_1 + (*u_{2,1} + \cdots + *u_{2,n})y_2 + \cdots + (*u_{n,1} + \cdots + *u_{n,n})y_n \\ \vdots \\ *y_1 + (*u_{2,1} + \cdots + *u_{2,n})y_2 + \cdots + (*u_{n,1} + \cdots + *u_{n,n})y_n \end{pmatrix}$$

Hence the coefficient of $y_1 y_i$ in $q_j((U+I_n)S_0^{-1}\mathbf{y})$ is linear in the variables $u_{k,l}$ when $i \neq 1$ and the coefficient of y_1^2 is a constant. □

[6] more precisely, generically non zero.

To summarize:

Lemma 6. *$I_{2,1}(F)$ contains exactly $n^2 + 2n$ linear equations.*

Proof. In $I_{2,1}(F)$, we have n linear equations from lemma 5, n linear equations from the very definition of $\mathcal{J}'$, and n^2 linear equations from lemma 5 □

As explained before, we obtain $n^2 + 2n$ linear equations for $I_{2,1}(F)$. However, we have $2n^2$ variables. So, we have to consider $I_{2,2}(F)$, *i.e.*, the equations generated at degree 2 during the second step. Thanks to lemma 6, we can reduce the original system to a quadratic system in $2n^2 - (2n+n^2) = (n-1)^2$ variables. W.l.o.g we can assume that we keep only the variable $u_{i,j}$ where $2 \leq i, j \leq n$. Let F' be the system obtained from F after substituting the $2n + n^2$ linear equations of lemma 6. All the monomials in $\mathbb{K}[x_1, \ldots, x_n, y_1, \ldots, y_n]$ of $\mathbf{Da}(S_0(S+I_n)\mathbf{x}, \mathbf{y}) - \mathbf{Db}(\mathbf{x}, (U+I_n)S_0^{-1}\mathbf{y})$ have the following shape:

$$x_i y_j y_k \text{ or } y_i x_j x_k \text{ with } 1 \leq i, j, k \leq n.$$

Hence the number of such monomials is $2n\frac{n(n+1)}{2} = n^2(n+1) \approx n^3$, which implies that the number of equations in F' is also n^3.

Thanks to this remark, we will now prove that we can linearize F'. Let $T(F')$ be the set of all monomials occurring in F'. We can assume that $T(G') = [t_1 < t_2 < \cdots < t_N]$. It is important to remark that $t_1 = u_{2,2}$ up to $t_{(n-1)^2} = u_{n,n}$ are in fact variables. Now, let M be the matrix representation of G' w.r.t. $T(G')$. Since we know precisely the shape of the equations from the proof of lemma 5, it is possible to establish that:

1. most of the equations are very sparse, namely each equation contains about n^2 non-zero terms.
2. all the variables $t_1, \ldots, t_{(n-1)^2}$ occur in *all* the equations

After a Gaussian elimination of the matrix M, we obtain the following shape:

$$\widetilde{M} = \begin{bmatrix} 1_{(n-1)^2} & 0 & 0 & 0 \\ 0 & \times & \cdots & \cdots \\ 0 & \times & \ddots & \vdots \\ 0 & \times & \cdots & \ddots \end{bmatrix}$$

Hence, we obtain after a second step of computation in degree 2 the equations $u_{2,2} = \cdots = u_{n,n} = 0$. This means that after 2 steps of computation at degree 2, we obtain $(n-1)^2 + 2n + n^2 = 2n^2$ linear equations in $2n^2$ unknowns. This explains why the maximum degree reached during the Gröbner basis computation of $\mathcal{I}'_{\mathbf{a},\mathbf{b}} + \mathcal{J}'$ is bounded by 2, and concludes the proof of theorem 3. □

5.1 Application to the Linear Inhomogeneous Case

If $c = 0$ in equation (1), and if $\mathbf{a}$ has a non-trivial homogeneous component of degree 1, then looking at the homogeneous component of degree one yields the image of S on one point. We are then in a situation where theorem 3 is applicable, and S can be determined though a Gröbner basis computation which terminates in time $\mathcal{O}\left(n^6\right)$.

5.2 Implementation and Application to the Other Cases

All the other cases reduce to the linear homogeneous case, as mentioned in section 2. In this setting, the problem is that we do not have enough knowledge on S to make the Gröbner basis computation efficient. A simple idea would be to guess a column of S then compute the Gröbner basis. This approach has complexity $\mathcal{O}\left(n^6 \cdot q^n\right)$ as explained before. It is possible to reduce this complexity by a factor of q, by discarding guesses for the column of S that yields different values of $\mathbf{a}$ and $\mathbf{b}$ on the corresponding points.

The biggest proposed cubic IP1S challenge (Challenge C in fig. 2) has $u = 1$, $n = 16$ and $q = 2$. Given one relation on S, the computation of the Gröbner basis takes 90 seconds on a 2.8Ghz Xeon computer using the publicly available implementation of F_4 in MAGMA. Since this has to be repeated 2^{15} times, the whole process takes about one CPU-month (and can be parallelized at will). For challenge D, the Gröbner basis is computed in 0.1 second, and the whole process takes about 2 hours.

5.3 An Interesting Failure

We conclude this section with a simple idea that could have lead to an improvement, by efficiently giving a relation on S, but which fails in an interesting manner. Let us denote by $Z_{\mathbf{a}}$ (resp. $Z_{\mathbf{b}}$) the set of zeroes of $\mathbf{a}$ (resp. $\mathbf{b}$). Because of lemma 1, and since S is linear, we have:

$$S\left(\sum_{\mathbf{x}\in Z_{\mathbf{a}}} \mathbf{x}\right) = \sum_{\mathbf{y}\in Z_{\mathbf{b}}} \mathbf{y}$$

This yields a relation on S, which is enough to use theorem 3. $\mathbf{a}$ and $\mathbf{b}$ may be assumed to have about q^{n-1} zeroes. Finding them requires time $\mathcal{O}\left(q^n\right)$. The complexity of the attack could thus be improved to $\mathcal{O}\left(n^6 + q^n\right)$. Surprisingly, this trick fails systematically, and this happen to be consequence of the Chevalley-Warning theorem [10,45].

Lemma 7. *The sum of the zeroes of a cubic form on 5 variables or more over $\mathbb{F}_q$ is always zero.*

Proof. Let us consider the elements of $Z_{\mathbf{a}}$ having α as their first coordinate, and let us denote by n_α their number. These are in fact the common zeroes of $(\mathbf{a}, x_1 - \alpha)$. By the Chevalley-Warning theorem [10,45], if $\mathbf{a}$ has at least 5 variables, then the characteristic of the field divides n_α. Therefore, their sum has zero on the first coordinate. Applying this result for all values of α shows that the sum of zeroes of $\mathbf{a}$ has a null first coordinate. We then just consider all coordinates successively. □

6 Conclusion

In this paper, we present algorithms for the IP problem with one secret for two random quadratic equations and one cubic equation. As already explained, there are the most cryptographically relevant instances. Moreover, we explain the complexity, success probability and give sufficient conditions so that the algorithms work. We combine the use of the differential and the computation of Gröbner bases of very overdefined systems. All the proposed IP1S challenges can be broken in practice by the technique we describe, as the following table shows.

Challenge	Time to break on one core
A	3 seconds
B	1 month
C	0 seconds
D	1 hours
E	3 minutes

In view of these results, we conclude that Patarin's IP1S-Based identification scheme is no longer competitive with respect to others combinatorial-based identification schemes [42,43,44,41].

References

1. Baena, J., Clough, C., Ding, J.: Square-vinegar signature scheme. In: PQCrypto '08: Proceedings of the 2nd International Workshop on Post-Quantum Cryptography, Berlin, Heidelberg, Springer-Verlag (2008) 17–30
2. Bardet, M., Faugère, J.C., Salvy, B., Yang, B.Y.: Asymptotic Behaviour of the Degree of Regularity of Semi-Regular Polynomial Systems. In: MEGA'05. (2005) Eighth International Symposium on Effective Methods in Algebraic Geometry, Porto Conte, Alghero, Sardinia (Italy), May 27th – June 1st.
3. Bardet, M.: Étude des systèmes algébriques surdéterminés. Applications aux codes correcteurs et à la cryptographie. PhD thesis, Université de Paris VI (2004)
4. Bardet, M., Faugère, J.C., Salvy, B.: On the complexity of Gröbner basis computation of semi-regular overdetermined algebraic equations. In: Proc. International Conference on Polynomial System Solving (ICPSS). (2004) 71–75
5. Bardet, M., Faugère, J.C., Salvy, B., Yang, B.Y.: Asymptotic behaviour of the degree of regularity of semi-regular polynomial systems. In: Proc. of MEGA 2005, Eighth International Symposium on Effective Methods in Algebraic Geometry. (2005)
6. Bennett, A.A.: Products of skew-symmetric matrices. American M. S. Bull. **25** (1919) 455–458
7. Bernstein, D.S.: Matrix mathematics. Theory, facts, and formulas. 2nd expanded ed. Princeton, NJ: Princeton University Press. xxxix, 1139 p. (2009)
8. Billet, O., Gilbert, H.: A traceable block cipher. In Laih, C.S., ed.: ASIACRYPT. Volume 2894 of Lecture Notes in Computer Science., Springer (2003) 331–346

9. Bosma, W., Cannon, J.J., Playoust, C.: The Magma Algebra System I: The User Language. J. Symb. Comput. **24**(3/4) (1997) 235–265
10. Chevalley, C.: Démonstration d'une hypothèse de M. Artin. Abh. Math. Semin. Hamb. Univ. **11** (1935) 73–75
11. Clough, C., Baena, J., Ding, J., Yang, B.Y., Chen, M.S.: Square, a new multivariate encryption scheme. In Fischlin, M., ed.: CT-RSA. Volume 5473 of Lecture Notes in Computer Science., Springer (2009) 252–264
12. Ding, J., Wolf, C., Yang, B.Y.: ℓ-invertible cycles for multivariate quadratic public key cryptography. In Okamoto, T., Wang, X., eds.: Public Key Cryptography. Volume 4450 of Lecture Notes in Computer Science., Springer (2007) 266–281
13. dit Vehel, F.L., Perret, L.: Polynomial Equivalence Problems and Applications to Multivariate Cryptosystems. In Johansson, T., Maitra, S., eds.: INDOCRYPT. Volume 2904 of Lecture Notes in Computer Science., Springer (2003) 235–251
14. Faugère, J.C., Gianni, P., Lazard, D., Mora, T.: Efficient Computation of Zero-Dimensional Gröbner Bases by Change of Ordering. Journal of Symbolic Computation **16**(4) (1993) 329–344
15. Faugère, J.C.: A new efficient algorithm for computing Gröbner bases (F4). Journal of Pure and Applied Algebra **139**(1-3) (June 1999) 61–88
16. Faugère, J.C.: A New Efficient Algorithm for Computing Gröbner Bases Without Reduction to Zero (F5). In: ISSAC '02: Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation, New York, NY, USA, ACM (2002) 75–83
17. Faugère, J.C., Perret, L.: Polynomial Equivalence Problems: Algorithmic and Theoretical Aspects. In Vaudenay, S., ed.: EUROCRYPT. Volume 4004 of Lecture Notes in Computer Science., Springer (2006) 30–47
18. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In Odlyzko, A.M., ed.: CRYPTO. Volume 263 of Lecture Notes in Computer Science., Springer (1986) 186–194
19. Fortin, S.: The graph isomorphism problem. Technical report, University of Alberta (1996)
20. Fuhrmann, P.A.: A polynomial approach to linear algebra. Springer-Verlag New York, Inc., New York, NY, USA (1996)
21. Fulman, J.: Random matrix theory over finite fields. Bull. Amer. Math. Soc. (N.S **39** 51–85
22. Garey, M.R., Johnson, D.S.: Computers and Intractability, A Guide to the Theory of *NP*-Completeness. Freeman, New-York (1979)
23. Geiselmann, W., Meier, W., Steinwandt, R.: An Attack on the Isomorphisms of Polynomials Problem with One Secret. Int. J. Inf. Sec. **2**(1) (2003) 59–64
24. Geiselmann, W., Steinwandt, R., Beth, T.: Attacking the Affine Parts of SFLASH. In Honary, B., ed.: IMA Int. Conf. Volume 2260 of Lecture Notes in Computer Science., Springer (2001) 355–359
25. Goldreich, O., Micali, S., Wigderson, A.: Proofs that yield nothing but their validity and a methodology of cryptographic protocol design (extended abstract). In: FOCS, IEEE (1986) 174–187
26. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof-systems (extended abstract). In: STOC, ACM (1985) 291–304
27. Koblitz, N.: Algebraic Aspects of Cryptography. Volume 3 of Algorithms and Computation in Mathematics. Springer-Verlag (1998)
28. Lazard, D.: Gröbner-bases, gaussian elimination and resolution of systems of algebraic equations. In van Hulzen, J.A., ed.: EUROCAL. Volume 162 of Lecture Notes in Computer Science., Springer (1983) 146–156
29. Lidl, R., Niederreiter, H.: Finite fields. Cambridge University Press, New York, NY, USA (1997)
30. Lyubashevsky, V.: Lattice-based identification schemes secure under active attacks. In Cramer, R., ed.: Public Key Cryptography. Volume 4939 of Lecture Notes in Computer Science., Springer (2008) 162–179
31. MacWilliams, J.: Orthogonal matrices over finite fields. The American Mathematical Monthly **76**(2) (1969) 152–164
32. Matsumoto, T., Imai, H.: Public quadratic polynomial-tuples for efficient signature-verification and message-encryption. In: Advances in Cryptology – EUROCRYPT 1988. Volume 330 of LNCS., Springer–Verlag (1988) 419–453
33. Naccache, D., ed.: Topics in Cryptology - CT-RSA 2001, The Cryptographer's Track at RSA Conference 2001, San Francisco, CA, USA, April 8-12, 2001, Proceedings. In Naccache, D., ed.: CT-RSA. Volume 2020 of Lecture Notes in Computer Science., Springer (2001)
34. Patarin, J.: Hidden fields equations (hfe) and isomorphisms of polynomials (ip): Two new families of asymmetric algorithms. In: EUROCRYPT. (1996) 33–48
35. Patarin, J.: Hidden Fields Equations (HFE) and Isomorphisms of Polynomials (IP): Two New Families of Asymmetric Algorithms. In: EUROCRYPT. (1996) 33–48 Etended version available on `http://www.minrank.org/hfe.pdf`.
36. Patarin, J.: The Oil and Vinegar signature scheme. presented at the Dagstuhl Workshop on Cryptography (1997)
37. Patarin, J., Courtois, N., Goubin, L.: Flash, a fast multivariate signature algorithm. [33] 298–307
38. Patarin, J., Courtois, N., Goubin, L.: QUARTZ, 128-Bit Long Digital Signatures. [33] 282–297
39. Patarin, J., Goubin, L., Courtois, N.: Improved Algorithms for Isomorphisms of Polynomials. In: EUROCRYPT. (1998) 184–200
40. Perret, L.: A Fast Cryptanalysis of the Isomorphism of Polynomials with One Secret Problem. In Cramer, R., ed.: EUROCRYPT. Volume 3494 of Lecture Notes in Computer Science., Springer (2005) 354–370
41. Pointcheval, D.: A new identification scheme based on the perceptrons problem. In: EUROCRYPT. (1995) 319–328
42. Shamir, A.: An efficient identification scheme based on permuted kernels (extended abstract). In Brassard, G., ed.: CRYPTO. Volume 435 of Lecture Notes in Computer Science., Springer (1989) 606–609

43. Stern, J.: A new identification scheme based on syndrome decoding. In Stinson, D.R., ed.: CRYPTO. Volume 773 of Lecture Notes in Computer Science., Springer (1993) 13–21
44. Stern, J.: Designing identification schemes with keys of short size. In Desmedt, Y., ed.: CRYPTO. Volume 839 of Lecture Notes in Computer Science., Springer (1994) 164–173
45. Warning, E.: Bemerkung zur vorstehenden Arbeit von Herrn Chevalley. Abh. Math. Semin. Hamb. Univ. **11** (1935) 76–83
46. Wolf, C., Preneel, B.: Taxonomy of Public Key Schemes Based on the Problem of Multivariate Quadratic Equations. Cryptology ePrint Archive, Report 2005/077 (2005)