The Security of the Birational Permutation Signature Schemes

Don Coppersmith
IBM Research
T. J. Watson Research Center
Yorktown Heights, NY 10598

Jacques Stern, Serge Vaudenay
Ecole Normale Supérieure
Laboratoire d'Informatique
45, rue d'Ulm, 75230 Paris

April 22, 1996

Abstract

In recent years, researchers have invested a lot of effort in trying to design suitable alternatives to the RSA signature scheme, with lower computational requirements. The idea of using polynomial equations of low degree in several unknowns, with some hidden trap-door, has been particularly attractive. One of the most noticeable attempt to push this idea forward is the Ong-Schnorr-Shamir signature scheme, which has been broken by Pollard and Schnorr. At Crypto'93, Shamir proposed a family of cryptographic signature schemes based on a new method. His design made subtle use of birational permutations over the set of k-tuples of integers modulo a large number N of unknown factorization. However, the schemes presented in Shamir's paper are weak. In the present paper, we describe several attacks which can be applied to schemes in this general family.

Introduction

The celebrated RSA cryptosystem can be viewed as a permutation computed in both directions as a polynomial over the ring $\mathbb{Z}_N$, where N is a (large) integer with secret factorization. In search for suitable alternatives to the RSA signature scheme, with lower computational requirements, several cryptographers have suggested to use polynomials of low degree in several variables. In the context of signature, such polynomials were natural candidates for the design of very efficient schemes, both for signature generation and signature verification.

The first cryptographic protocol based on this principle is the Ong-Schnorr-Shamir signature scheme [5]. It has been broken by Pollard and Schnorr [7]. At Crypto'93, Shamir proposed a family of cryptographic signature schemes based on a new method. His design made subtle use of birational permutations of the integers modulo N. Shamir actually introduced several techniques: the first technique uses as a trap-door a family of quadratic forms built in a very specific way and which he calls *sequentially linearized*. It is a kind of generalization of the Ong-Schnorr-Shamir scheme, with more unknowns and more equations to be solved for signature generation. Another technique uses the notion of an algebraic basis for the quadratic forms: this is a set of quadratic forms from which any other one can be computed by using only rational operations. This technique can be further divided according to the algebraic basis chosen and Shamir's paper includes two proposals, one using a symmetric basis and the other an asymmetric one. Of course, there is nothing specific to quadratic forms in Shamir's approach: it only turns out that use of cubic or quartic polynomials makes key management cumbersome and loses the computational advantages shown by the scheme.

In the present paper, we show that the schemes presented in Shamir's paper are weak, by exhibiting several attacks which can be applied to schemes in the general family. These results have been announced in [2], where we deal with the trapdoor based on sequentially linearized equations and with the symmetric basis proposal. Since then, another attack has appeared in [10], which takes care of the asymmetric basis.

It is worth mentioning that another public key system scheme based on quadratic forms has been proposed by Matsumoto and Imai [4]. This scheme is based on completely different ideas and uses (small) fields of characteristic 2. Let us add that the Matsumoto-Imai scheme has recently been broken by Patarin [6]. Thus, there seems to be some kind of intrinsic difficulty that prevents hiding trap-doors into families of quadratic forms.

We close this introduction by thanking Adi Shamir both for sending us his Crypto'93 paper at an early stage and for many discussions on the subject of this paper.

1 The methodology of the attacks.

1.1 The overall strategy.

Basically, Shamir's idea is to start from a family of quadratic forms with some "visible" algebraic structure (e.g. low rank) and to hide the underlying structure by performing the following operations

1. linear change of coordinates
2. linear combinations of the resulting forms

We are thus faced to the problem of trying to recapture some of the hidden structure, from the public key only. This public key consists of several quadratic forms and we note that, as a consequence of step 2 above, some linear combinations of the public forms may retain a part of the original algebraic structure. Unfortunately, we can only handle these objects indirectly, through the use of indeterminate coefficients, say $\delta, \epsilon, \ldots$. At this point we note that many of the properties used in the design proposed by Shamir can be expressed by the vanishing of polynomials in $\delta, \epsilon, \ldots$. We quote several examples:

- the fact that a quadratic form has not full rank is expressed by the vanishing of its determinant
- the fact that a quadratic form has rank 2 is expressed by the vanishing of all 3×3 determinants
- the fact that a vector u belongs to the vector space spanned by the rows of the matrix M of a given quadratic form of rank k is expressed by the vanishing of all $(k+1) \times (k+1)$ determinants of the matrix M' obtained by appending u as an extra row to M. These determinants are polynomials in $\delta, \epsilon, \ldots$ and in the coordinates of u.

1.2 Galois theory and ideal calculations.

We are thus led to a set of polynomial equations in $\delta, \epsilon, \ldots$. Such a set of equations generates an ideal in the ring of polynomial with several unknowns: in other words, if $P_1, \ldots, P_m$ are m polynomials with unknowns $\delta, \epsilon, \ldots$, the equations $P_i = 0$ define an algebraic curve associated to the ideal of all polynomials which can be written

$$P_1 Q_1 + \ldots + P_m Q_m$$

where $Q_1, \ldots, Q_m$ are arbitrary polynomials.

At this point, we have to return to the underlying structure. In case there is a lack of symmetry, as in Shamir's first scheme, we can try to solve for one of the unknowns: this simply means that the ideal should contain a polynomial of degree one with a single variable. In other cases, we observe a strong symmetry: for example, in the scheme based on the symmetric basis, we isolate a sequence of integers modulo N, say $\delta_1, \ldots, \delta_k$, coming from the hidden structure, which act as (say) first coordinates of points $m_1, \ldots, m_k$ of the curve which cannot be distinguished from each other. In such a case, it is hopeless to try to solve for the first coordinate δ. On the other hand, we expect to find in the ideal a polynomial of degree k in the single variable δ, $F(\delta)$, which we can treat symbolically and of which the values $\delta_1, \ldots, \delta_k$ are the unknown roots. This is a context close to Galois theory. Still, we do not really offer proofs of the various statements we make relying on Galois-like arguments. Although it might be possible to write up proofs in some cases, we feel that the technicalities would distract from the issues at hand. In place, we remain at an informal level and implicitly assume a large degree of "genericity". We think that this is perfectly acceptable in a paper concerned with cryptanalysis: furthermore, our attack has been implemented using a computer algebra package, and this is a kind of experimental verification of the correctness of our statements.

We now turn to ideal calculations. As explained above, we need to disclose members of the ideal with prescribed degrees for the various unknowns. For this, we can use Gröbner basis algorithms (see [1]), which output another family of polynomials $P'_1, \ldots, P'_r$ spanning the same ideal and which is *reduced* in a suitable sense. The drawback of this algorithm is its high complexity. In case we are trying to eliminate all unknowns except one, we can repeatedly form resultants of two polynomials with respect to a given unknown. We can also apply the Euclidean algorithm to compute the g.c.d. of two polynomials with respect to to a given unknown. This decreases the degree of an equation.

Finally, it is also possible to use a simpler *ad hoc* version of the Gröbner basis algorithm which is a kind of generalized Gaussian elimination. For instance, if M is a monomial in P_1 which does not divide any other monomial of P_1, then every multiple monomial of M can be eliminated in the other polynomials by replacing P_i by $P'_i = P_i - Q_i P_1$ for a suitable polynomial Q_i. Then, one can continue the reduction with $P'_2, \ldots, P'_m$. In most cases, if m is large enough and if there is a hidden trap-door in the set of equations, this reduction is likely to end rather quickly with very simple equations of the expected form. In the rest of the paper, we will not comment further on ideal calculations and will thus treat them as a kind of "programming technology", which we actually used in our experiments.

1.3 Working mod N versus working mod p.

Our analysis basically treats the ring of integers mod N as a field. Actually, N is composite and we assume for simplicity that it has only two prime factors p and q. Our calculations make sense mod p since we are actually working in a field but some justification is needed to go from calculations mod p to calculations mod N. In section 2, we will only use tools from linear algebra such as Gaussian elimination or determinants. Thus all computations go through regardless of the fact that N is composite. The situation is a bit more subtle in section 3, where Galois theory comes into the picture. For instance, assume that we have discovered a polynomial $F(\delta)$ of degree k from a sequence of integers modulo N, say $\delta_1, \ldots, \delta_k$, coming from the hidden structure, as explained in section 1.2. Such a polynomial has k solutions mod p but k^2 solutions mod N, each obtained by mixing some solution mod p with some solution mod q. But if we consider only the image, mod p, of our calculations mod N, things are all right. As will be shown, our cryptanalysis provides a way to forge signatures by performing calculations which treat δ (and possibly other variables) symbolically. Galois-like arguments show that the result has the expected symmetry and thus, is expressible in terms of the coefficients of F and in terms of the coefficients of the public key. These calculations are valid mod p. They are also valid mod q, and the Chinese remainder theorem suffices to make them valid mod N. This is in spite of the fact that a solution δ of $F \bmod N$ might well mix different solutions $\delta_i \bmod p$ and $\delta_j \bmod q$. Since we never explicitly solve for δ, but only work with it symbolically and use the fact that $F(\delta) = 0 \bmod N$, we never are in danger of factoring N.

2 The first scheme

The first family of Shamir's signature schemes is based on sequentially linearized equations. The public information consists of a large integer N of unknown factorization (even the legitimate users need not know its factorization), and the coefficients of $k-1$ quadratic forms $f_2, \ldots, f_k$ in k variables $x_1, \ldots, x_k$ each. Each of these quadratic forms can be written as

$$f_i = \sum_{j,\ell} \alpha_{ij\ell} x_j x_\ell \tag{1}$$

where i ranges from 2 to k and the matrix $\alpha_{ij\ell}$ is symmetric i.e. $\alpha_{ij\ell} = \alpha_{i\ell j}$.

The secret information is a pair of linear transformations. One linear transformation B relates the quadratic forms $f_2, \ldots, f_k$ to another sequence of quadratic forms $g_2, \ldots, g_k$. The second linear transformation A is a change of coordinates that relates the variables $(x_1, \ldots, x_k)$ to a set of "original" variables $(y_1, \ldots, y_k)$. Denoting by Y the column vector of the original variables and by X the column vector of the new variables, we can simply write $Y = AX$.

Of course, the coefficients of A and B are known only to the legitimate user. The trap-door requirements are twofold: when expressed in terms of the original variables $y_1, \ldots, y_k$, the quadratic form g_2 is computed as:

$$g_2 = y_1 y_2 \tag{2}$$

and the subsequent g_i's, $3 \leq i \leq k$ are *sequentially linearized*, i.e. can be written

$$g_i(y_1, \ldots, y_k) = \ell_i(y_1, \ldots, y_{i-1}) \times y_i + q_i(y_1, \ldots, y_{i-1}) \tag{3}$$

where ℓ_i is a linear function of its inputs and q_i is a quadratic form.

To sign a message M, one hashes M to a $k-1$-tuple $(f_2, \ldots, f_k)$ of integers modulo N, then finds a sequence $(x_1, \ldots, x_k)$ of integers modulo N satisfying (1). This is easy from the trap-door.

It is straightforward that the particular case $k = 2$ is equivalent to the Ong-Schnorr-Shamir scheme [5]. The Pollard-Schnorr algorithm [7] enables to forge a valid signature of any message. In the following, we show how to break the other cases reducing them to the Ong-Schnorr-Shamir scheme too.

We let M_i, $2 \leq i \leq k$ denote the $k \times k$ symmetric matrix of the quadratic form g_i. The kernel K_i of g_i is the kernel of the linear mapping whose matrix is M_i. It consists of vectors which are orthogonal to all vectors with respect to g_i. The rank of the quadratic form g_i is the rank of M_i. It is the codimension of K_i as well as the unique integer r such that g_i can be written as a linear combination of squares of r independent linear functionals. (For more details, see [3] for instance.) Actually, all this is not completely accurate as N is not a prime number and therefore $\mathbb{Z}_N$ is not a field. This question has been addressed in section 1.3 and we now ignore the problem.

An easy computation shows that K_i is the subspace defined in terms of the original variables by the equations

$$y_1 = \ldots = y_i = 0 \tag{4}$$

From this, it follows that
i) K_i is decreasing;
ii) the dimension of K_i is $k - i$;
iii) any element of K_{i-1} not in K_i is an isotropic element wrt g_i, which means that the value of g_i is zero at this element.

We will construct a basis b_i of the k-dimensional space, such that the family $b_{i+1}, \ldots, b_k$ spans K_i for $i = 2, \ldots, k-1$. The main problem we face is the fact that the g_i's and therefore the K_i's are unknown. Instead, we know the f_i's. We concentrate on the (unknown) coefficient δ_i of g_k in the expression of f_i, i.e. we write

$$f_i = \delta_i g_k + \sum_{j=2}^{k-1} \beta_{ij} g_j \tag{5}$$

As coefficients have been chosen randomly, we may assume that δ_k is not zero. Let $i < k$. Consider the quadratic form (in all x_is) $Q_i(\lambda) = f_i - \lambda f_k$. When $\lambda = \delta_i/\delta_k$, this form has a non-trivial kernel and therefore δ_i/δ_k is a root of the polynomial $P_i(\lambda) = \det(Q_i(\lambda))$. This is not enough to recover the correct value of λ. Computing the matrix of $Q_i(\lambda)$ for $\lambda_i = \delta_i/\delta_k$ in the basis corresponding to the original coordinates $y_1, \ldots, y_k$ yields the following

$$\begin{pmatrix} \boxed{C} & \boxed{0} \\ \boxed{0} & 0 \end{pmatrix}$$

In the same basis, the matrix of $Q_i(\lambda)$ for any λ, can be written as

$$\begin{pmatrix} \boxed{C_\lambda} & \boxed{U_\lambda} \\ \boxed{(U_\lambda)^t} & 0 \end{pmatrix}$$

We observe that U_λ is affine in λ and vanishes at λ_i so that the determinant of the matrix is divisible by $(\lambda - \lambda_i)^2$. Since determinants can be computed up to a multiplicative constant in any basis, it follows that $(\lambda - \lambda_i)^2$ factors out in $P_i(\lambda)$. Thus the correct value of λ_i can be found by observing that it is a double root of the polynomial equation $P_i(\lambda) = 0$. We now make use of the informal genericity principle explained in section 1.2, which means that we ignore "exceptional" situations. As a consequence, we claim that the double root is disclosed by simply taking the g.c.d. in $\mathbb{Z}_N$ of P_i and P'_i with respect to λ. We find a linear equation in λ, from which we easily compute λ_i.

Once all coefficients λ_i have been recovered, we set for $i = 2, \ldots, k-1$

$$\tilde{f}_i = f_i - \lambda_i f_k \tag{6}$$

and $\tilde{f}_k = f_k$. We note that all quadratic forms $\tilde{f}_i$ have kernel K_{k-1}. This allows to pick a non-zero vector b_k in K_{k-1}. The construction can then go on inductively in the quotient space of the k-dimensional space by the vector spanned by $\{b_k\}$ with $\tilde{f}_2, \ldots, \tilde{f}_{k-1}$ in place of $f_2, \ldots, f_k$.

At the end of the recursive construction, we obtain a sequence b_i, $3 \leq$

$i \leq k$ such that $b_{i+1}, \ldots, b_k$ spans K_i for $i = 2, \ldots, k-1$ and a sequence of quadratic forms $\tilde{f}_2, \ldots, \tilde{f}_k$ such that
i) $\tilde{f}_i$ *has kernel* K_i*;*
ii) b_i *is an isotropic element wrt* $\tilde{f}_i$*.*
Choosing b_1, b_2 at random, we get another set of coordinates $z_1, \ldots, z_k$ defined by $X = (b_1 \ldots b_n)Z$ such that
i) $\tilde{f}_2$ *is a quadratic form in the coordinates* z_1, z_2
ii) $\tilde{f}_3, \ldots, \tilde{f}_k$ *is sequentially linearized*

The rest is easy. From a sequence of prescribed values for $f_2, \ldots, f_k$, we can compute the corresponding values of $\tilde{f}_2, \ldots, \tilde{f}_k$. Next, we can find values of $\{z_1, z_2\}$ achieving a given value of $\tilde{f}_2$ mod N in exactly the same way as the Pollard solution of the Ong-Schnorr-Shamir scheme [5]. Then, values for $z_3, \ldots, z_k$ achieving given values of $\tilde{f}_3, \ldots, \tilde{f}_k$ are found by successively solving $k-2$ linear equations. Finally, the values of $z_1, \ldots, z_k$ can be translated into values of $x_1, \ldots, x_k$.

Example. In Shamir's paper [9], an example is given with $N = 101$ (the fact that 101 is prime is unfortunate but actually irrelevant).

$$v_2 = 78x_1^2 + 37x_2^2 + 6x_3^2 + 54x_1x_2 + 19x_1x_3 + 11x_2x_3 \pmod{101}$$

$$v_3 = 84x_1^2 + 71x_2^2 + 48x_3^2 + 44x_1x_2 + 33x_1x_3 + 83x_2x_3 \pmod{101}.$$

Matrices of f_2, f_3 are as follows

$$\begin{pmatrix} 78 & 27 & 60 \\ 27 & 37 & 56 \\ 60 & 56 & 6 \end{pmatrix} \qquad \begin{pmatrix} 84 & 22 & 67 \\ 22 & 71 & 92 \\ 67 & 92 & 48 \end{pmatrix}$$

We get:

$$P(\lambda) = \det(f_2 - \lambda f_3) = 34(\lambda^3 + 75\lambda^2 + 55\lambda + 71) \tag{7}$$

$$P'(\lambda) = \lambda^2 + 50\lambda + 52 \tag{8}$$

$$\gcd(P, P') = \lambda - 63 \tag{9}$$

We let

$$\tilde{f}_2 = f_2 - 63 f_3 \quad ; \quad \tilde{f}_3 = f_3 \tag{10}$$

The kernel of $\tilde{f}_2$ is spanned by vector $b_3 = (31, 12, 1)^t$. We pick $b_2 = (0, 1, 0)^t$ and $b_1 = (1, 31, 0)^t$. We get, in the corresponding coordinates z_1, z_2, z_3:

$$\tilde{f}_2 = 26 z_1^2 + 8 z_2^2 \quad ; \quad \tilde{f}_3 = z_3(26 z_1 + 20 z_2) + 90 z_1^2 + 2 z_1 z_2 + 71 z_2^2 \tag{11}$$

Then, for any tuple (f_2, f_3) of integers, we compute $(\tilde{f}_2, \tilde{f}_3)$ using (10), we solve $\tilde{f}_2 = 26 z_1^2 + 8 z_2^2$ by Pollard-Schnorr's algorithm and compute z_3 such that equations (11) hold. This forges a signature.

3 The second scheme

We now treat Shamir's [9] second scheme. Throughout, we will pretend we are working in $\mathbb{Z}_p$ rather than $\mathbb{Z}_N$. This has been explained in section 1.3.

We briefly review the scheme. Shamir begins with k variables $y_1, y_2, \ldots, y_k$, with k odd. These are subjected to a secret linear change of variables which gives $u_i = \sum_j a_{ij} y_j, i = 1, 2, \ldots, k$, with the matrix $A = (a_{ij})$ secret. The products $u_i u_{i+1}$, including $u_k u_1$, are subjected to a second secret linear transformation $B = (b_{ij})$, so that $v_i = \sum_j b_{ij} u_j u_{j+1}, i = 1, 2, \ldots, k - s$. The public key is the set of coefficients $(c_{ij\ell})$ expressing v_i in terms of pairwise products $y_j y_\ell$, for $1 \le i \le k - s$,

$$v_i = \sum_{j,\ell} c_{ij\ell} y_j y_\ell, 1 \le i \le k - s, c_{ij\ell} = c_{i\ell j} \tag{12}$$

In the above, $s \ge 1$ is a parameter and, for the sake of simplicity, we treat first the case $s = 1$. Thus, i is ranging to $k - 1$, meaning that we have discarded $s = 1$ of the v_i. A valid signature of a $(k-1)$-tuple of integers $(v_1, \ldots, v_{k-1})$ is a set of values of $y_1 y_2, \ldots, y_k y_1$ such that (12) holds. Signature generation for the legitimate user is based on the fact that $y_1 y_2, \ldots, y_k y_1$ form an algebraic basis for the ideal generated by quadratic forms: for example, if $k = 3$, y_1^2 is recovered by the formula

$$y_1^2 = \frac{(y_1 y_2)(y_3 y_1)}{y_2 y_3}$$

See [9] for more details.

The first step in our solution is as follows: linear combinations of the v_i are linear combinations of the $u_i u_{i+1}$, but they form only a subspace of dimension $k-1$. Some linear combinations of the v_i,

$$v_1 + \delta v_2 + \sum_{3 \le j \le k-1} \epsilon_j v_j \tag{13}$$

will be quadratic forms in the y_i of rank 2. A computation shows that the only linear combinations of the products $u_i u_{i+1}$ of rank 2 are of the form

$$\alpha_i u_{i-1} u_i + \beta_i u_i u_{i+1} = u_i(\alpha_i u_{i-1} + \beta_i u_{i+1}) \tag{14}$$

for any values of α_i, β_i, i. Because the v_j span a subspace of codimension 1, and because we are further restricting to one lower dimension by the choice of the multiplier 1 for v_1 in the linear combination, we find that for each i there will be one pair (α_i, β_i) and one set of coefficients $(\delta_i, \epsilon_{ij})$ such that

$$\alpha_i u_{i-1} u_i + \beta_i u_i u_{i+1} = u_i(\alpha_i u_{i-1} + \beta_i u_{i+1}) = v_1 + \delta_i v_2 + \sum_{3 \le j \le k-1} \epsilon_{ij} v_j \tag{15}$$

We now omit the i indices for the sake of clarity. The condition of being rank 2 is an algebraic condition: setting

$$v_1 + \delta v_2 + \sum_{3 \le j \le k-1} \epsilon_j v_j = \sum_{j\ell} \tau_{j\ell} y_j y_\ell \tag{16}$$

with $\tau_{j\ell} = \tau_{\ell j}$, we find that each 3×3 submatrix of the matrix $(\tau_{j\ell})$ has vanishing determinant. Each of these determinants is a polynomial equation in δ, ϵ_j. Use resultants and Gaussian elimination to eliminate ϵ_j from this family of polynomial equations (in the ring $\mathbb{Z}_N$) and find a single polynomial F of degree k satisfied by δ. We also find ϵ_j as polynomials in δ, by returning to the original equations and eliminating the variables $\epsilon_i, i \ne j$.

Thus each solution δ to $F(\delta) = 0$ gives rise to a linear combination of v_j which is of rank 2. The root δ corresponds to that index i for which

$$v_1 + \delta v_2 + \sum_{3 \le j \le k-1} \epsilon_j v_j = u_i(\alpha_i u_{i-1} + \beta_i u_{i+1}) \tag{17}$$

We will indicate this correspondence by writing $\delta = \delta_i$.

For each solution $\delta = \delta_i$, the rows of the resulting matrix (τ_{ij}) span a subspace $Y(\delta_i) = Y_i$ of ${\mathbb{Z}_p}^k$ of rank 2; namely, Y_i is spanned by u_i and

$\alpha_i u_{i-1} + \beta_i u_{i+1}$. This observation is rather straightforward if the quadratic form is expressed in the basis corresponding to the u_i variables. Going to the y_i coordinates involves a right multiplication by A and a left multiplication by its transpose A^t. The first operation replaces row vectors by their expressions in terms of the "new" variables y_i and the latter does not affect the vector space spanned by the rows. This is enough to conclude.

Observe that u_i, u_{i+2}, and $(\alpha_{i+1}u_i + \beta_{i+1}u_{i+2})$ are linearly related, as are u_i, u_{i-2}, and $(\alpha_{i-1}u_{i-2} + \beta_{i-1}u_i)$. So

$$u_i \in Y_i \cap (Y_{i+1} + Y_{i+2}) \cap (Y_{i-1} + Y_{i-2}) \tag{18}$$

This is an algebraic relation among δ_{i-2}, δ_{i-1}, δ_i, δ_{i+1}, and δ_{i+2}. More accurately, for $k > 5$, $(i+1, i+2)$ and $(i-1, i-2)$ are the only instances of pairs (a, b) (c, d) consisting of four different indices, all distinct from i, such that

$$Y_i \cap (Y_a + Y_b) \cap (Y_c + Y_d) \neq \{0\} \tag{19}$$

We thus introduce five different variables δ, δ', δ'' etc., representing δ_i, δ_{i+1}, δ_{i+2}, δ_{i-1} and δ_{i-2}, and we formulate the relation as the vanishing of several determinants, as explained in section 1.1. We then reduce the resulting ideal by factoring out any occurrences of $(\delta - \delta')$, $(\delta - \delta'')$, etc. to assure that δ, δ' etc. are really different solutions. That is, we consider the ideal formed by $F(\delta)$, $F(\delta')$ etc. $(F(\delta) - F(\delta'))/(\delta - \delta')$, etc. and the various determinants derived from 19, and we apply the Gröbner basis reduction or the Euclidean algorithm to this ideal to find a basis.

Only multiples of some u_i satisfy such a relation (18) over $\mathbb{Z}_p$. We fix a multiple of each u_i by normalizing u_i to have first coordinate 1. The relations finally serve to define u_i in terms of δ_i.

By a similar argument, there is a quadratic equation $G(\delta_i, \delta_{i+1})$ expressing δ_{i+1} in terms of δ_i, whose two solutions are δ_{i+1} and δ_{i-1}. For $k > 5$, the algebraic condition is that the corresponding spaces Y_i, Y_{i+1} are in two different triples of subspaces enjoying linear relations:

$$\operatorname{rank}(Y_i + Y_{i+1} + Y_{i+2}) = \operatorname{rank}(Y_i + Y_{i+1} + Y_{i-1}) = 5 \tag{20}$$

Special considerations for small k. For $k = 5$, the above arguments do not apply since there are more instances of the relation

$$Y_i \cap (Y_a + Y_b) \cap (Y_c + Y_d) \neq \{0\}$$

with distinct (a, b, c, d, i). For example

$$Y_1 \cap (Y_2 + Y_5) \cap (Y_3 + Y_4)$$

is a one-dimensional space spanned by a vector of the form $u_5 + \mu u_2$ for some constant μ. It turns out that a pair of adjacent spaces such as $(Y_3 + Y_4)$ appear in three such relations whereas a pair of non-adjacent spaces such as $(Y_2 + Y_5)$ appears only in one.This gives an algebraic condition to identify pairs of adjacent δ_is. Once this is done, we can use a two pairs of adjacent indices (a, b), (c, d), with a, b, c, d distinct and different from i and the relation

$$u_i \in Y_i \cap (Y_a + Y_b) \cap (Y_c + Y_d)$$

in order to express u_i as a function of δ_i of degree 4.

The particular case $k = 3$ deserves independent discussions and is postponed until section 4.

Returning to the general case $k \geq 5$, we represent the solution of the quadratic equation by τ, and say that (δ, τ) generates a pair of 'adjacent' elements (u_i, u_{i+1}) (elements which are multiplied together in the original signature). We think of δ as generating an extension of degree k over $\mathbb{Z}_N$, and τ as generating an extension of degree 2 over $\mathbb{Z}_N[\delta]/F(\delta)$. The ability to distinguish the unordered pairs of 'adjacent' roots $\{\delta_i, \delta_{i+1}\}$ makes the system similar, in spirit, to a Galois extension of $\mathbb{Q}$ whose Galois group is the dihedral group on k elements. We will call on this analogy later. (Remark: it is only an analogy, because δ and τ really are elements of the ground fields.)

We can get the missing kth equation

$$v'_k = \sum_i u_i u_{i+1} \tag{21}$$

The coefficients of v'_k in terms of $y_j y_\ell$ ostensibly depend on δ_i and on the pairings (δ_i, δ_{i+1}), or equivalently on (δ, τ). But the coefficients would come out the same no matter which solution (δ, τ) were chosen, that is, no matter whether we assigned the ordering $(1, 2, 3, \ldots, k)$ or $(3, 2, 1, k, k-1, \ldots, 4)$ to the solutions u_i. This means that the coefficients will be in fact independent of (δ, τ). They will be expressible in terms of only the coefficients of the original $v_i, 1 \leq i \leq k$. This is because they are symmetric (up to dihedral symmetry) in the solutions δ_i.

The arguments here are analogous to those of Galois theory. Each coefficient c of v'_k is expressed as

$$c = \sum_{0 \leq i < k, 0 \leq j \leq 1} w_{ij} \delta^i \tau^j \tag{22}$$

For each of $2k$ different choices of (δ, τ) the value of c comes out the same. Treating (22) as $2k$ linear equations in the $2k$ unknowns w_{ij}, with coefficients given by $\delta^i \tau^j$ for various choices of (δ, τ), we must find (if the matrix has full rank) that $w_{00} = c$, and $w_{ij} = 0$ for $(i, j) \neq (0, 0)$.

Now we wish to solve a particular signature. We are given the integer values $v_1, \ldots, v_{k-1}$, and we assign an arbitrary value to v'_k. We have the equations relating v_i to $u_j u_{j+1}$:

$$v_i = \sum_j b'_{ij} u_j u_{j+1} \tag{23}$$

where b'_{ij} depends on δ_j. Select (symbolically) one pair (δ, τ) to fix the first two solutions (u_1, u_2), and compute the others in terms of (δ, τ). Then we have $b'_{ij} u_j u_{j+1}$ depending only on (δ, τ).

At this point we have v'_k (which is a v_k-like quadratic form), and A-like and B-like matrices respectively denoted by A' and B' (expressing linear transformations); the entries of all of these live in the pseudo Galois extension $\mathbb{Z}_N[\delta, \tau]/F(\delta)/G(\delta, \tau)$ expressing linear transformations. All rational operations can be done in this domain so this enables us to sign in it. Since the resulting signature does not depend on the ordering of the δ_i, its coefficients will always be in $\mathbb{Z}_N$. Thus it is possible to forge any signature working in a more complicated domain and getting results which always end up in $\mathbb{Z}_N$.

The attack has been implemented on a Sparc Workstation using the computer algebra system MAPLE. It computes a secret key-like (v'_k, A', B') within few hours and then forges any signature in a negligible time.

Example. As for the first scheme, we give a toy example with $N = 97$, $k = 5$ and $s = 1$. The public key is as follows:

$$\begin{aligned} v_1 \;&=\; 11y_1^2 + 31y_2^2 + 15y_3^2 + 8y_4^2 + 5y_5^2 + 23y_1y_2 + 89y_1y_3 + 60y_1y_4 + \\ &\quad\; 47y_1y_5 + 43y_2y_3 + 24y_2y_4 + 93y_2y_5 + 9y_3y_4 + 78y_3y_5 + 32y_4y_5 \end{aligned}$$

$$
\begin{aligned}
v_2 &= 83y_1^2 + 32y_2^2 + 16y_3^2 + 13y_4^2 + 92y_5^2 + 28y_1y_2 + 83y_1y_3 + 58y_1y_4 + \\
&\quad 84y_1y_5 + 58y_2y_3 + 64y_2y_4 + 84y_2y_5 + 38y_3y_4 + 69y_3y_5 + 36y_4y_5 \\
v_3 &= 45y_1^2 + 33y_2^2 + 96y_3^2 + 75y_4^2 + 90y_5^2 + 34y_1y_2 + 51y_1y_3 + 89y_1y_4 + \\
&\quad 26y_1y_5 + 16y_2y_3 + 90y_2y_4 + 42y_2y_5 + 9y_3y_4 + 8y_3y_5 + 47y_4y_5 \\
v_4 &= 65y_1^2 + 54y_2^2 + 96y_3^2 + 33y_4^2 + 26y_5^2 + 46y_1y_2 + 25y_1y_3 + 75y_1y_4 + \\
&\quad 76y_1y_5 + 59y_2y_3 + 66y_2y_4 + 95y_2y_5 + 69y_3y_4 + 48y_3y_5 + 56y_4y_5
\end{aligned}
$$

For the sake of brevity, we did not include the secret key, since we will show how to sign, given the public key only. As far as signature verification is concerned, we propose, as an example, a valid signature of the message which hashes onto $(1, 2, 3, 4)$, namely.

$$y_1y_2 = 7 \ , \ y_2y_3 = 92 \ , \ y_3y_4 = 69 \ , \ y_4y_5 = 54 \ , \ y_5y_1 = 70$$

From these values, one can compute all corresponding values of y_iy_j and check that $(v_1, v_2, v_3, v_4) = (1, 2, 3, 4)$.

As explained above, we consider the quadratic form

$$v_1 + \delta v_2 + \epsilon_3 v_3 + \epsilon_4 v_4$$

and we express the vanishing of all its (3×3) minors. As an example, one of the determinants provides the following equation:

$$
\begin{aligned}
&7\delta + 15\epsilon_4 + 83 + 18\delta^2 + 31\delta\epsilon_3 + 69\delta\epsilon_4 + 17\epsilon_3^2 + 71\epsilon_4^2 + 32\epsilon_3\epsilon_4 + \\
&49\delta^2\epsilon_3 + 40\delta^2\epsilon_4 + 47\delta\epsilon_3^2 + 54\delta\epsilon_4^2 + 78\epsilon_3^2\epsilon_4 + \epsilon_3\epsilon_4^2 + 24\epsilon_4^3 + 50\epsilon_3^3 + 38\delta^3 + \\
&67\delta\epsilon_3\epsilon_4 + 5\epsilon_3 = 0.
\end{aligned}
$$

Using reduction, we get:

$$
\begin{aligned}
F(\delta) &= 92 + 58\delta + 51\delta^2 + 43\delta^3 + 72\delta^4 + \delta^5 \\
\epsilon_3 &= 44 + 29\delta + 83\delta^2 + 95\delta^3 + 56\delta^4 \\
\epsilon_4 &= 87 + 14\delta + 94\delta^2 + 33\delta^3 + 38\delta^4
\end{aligned}
$$

Next, we make use of the ideal generated by all $F(\delta)$, $F(\delta')$, $(F(\delta) - F(\delta'))/(\delta - \delta')$ etc. and by the polynomials expressing that two roots are adjacent. Using τ to denote a root adjacent to δ and reducing the ideal, we get, as expected, an equation of degree two w.r.t. τ, say $G(\delta,\tau)$ and we compute the other roots in terms of δ and τ. This yields:

$$\begin{aligned}
G(\delta,\tau) &= 67\delta^4\tau + 20\delta^4 + 89\delta^3\tau + 26\delta^3 + 68\delta^2\tau + 85\delta^2 + 6\delta\tau + \\
&\quad 3\delta + \tau^2 + 43\tau + 57 \\
\delta_3 &= 68\delta^4\tau + 23\delta^4 + 90\delta^3\tau + 86\delta^3 + 85\delta^2\tau + 18\delta^2 + 93\delta\tau + \\
&\quad 35\delta + 42\tau + 93 \\
\delta_4 &= 29\delta^4\tau + 44\delta^4 + 7\delta^3\tau + 3\delta^3 + 12\delta^2\tau + 50\delta^2 + 4\delta\tau + 67\delta + \\
&\quad 55\tau + 72 \\
\delta_5 &= 30\delta^4 + 8\delta^3 + 29\delta^2 + 91\delta + 96\tau + 54
\end{aligned}$$

We also compute the values of all normalized u_is in terms of δ and τ from equations (18). As an example, here is the output for u_2:

$$\begin{aligned}
u_2 &= y_1 + \\
&\quad y_2(85\delta^4\tau + 89\delta^4 + 15\delta^3\tau + 87\delta^3 + 38\delta^2\tau + 88\delta^2 + 69\tau\delta + 6\delta + 35\tau + 12) + \\
&\quad y_3(86\delta^4\tau + 31\delta^4 + 41\delta^3\tau + 13\delta^3 + 24\delta^2\tau + 18\delta^2 + 52\delta\tau + 62\delta + 15\tau + 17) + \\
&\quad y_4(43\delta^4\tau + 45\delta^4 + 52\delta^3\tau + 38\delta^3 + 68\delta^2\tau + 58\delta^2 + 2\delta\tau + 88\delta + 27\tau + 87) + \\
&\quad y_5(28\delta^4\tau + 4\delta^4 + 8\delta^3\tau + 75\delta^3 + 74\delta^2\tau + 73\delta^2 + 45\delta + 29\delta\tau + 58\tau + 75).
\end{aligned}$$

All computations now take place in the pseudo Galois extension $\mathbb{Z}_N[\delta,\tau]/F(\delta)/G(\delta,\tau)$. We choose v_5 as the sum of all u_iu_{i+1}. As expected, this value turns out to be "independent" of δ and τ:

$$\begin{aligned}
v_5 &= 5y_1^2 + 30y_2^2 + 89y_3^2 + 67y_4^2 + 5y_5^2 + 35y_1y_2 + 4y_1y_3 + 62y_1y_4 + 61y_1y_5 + \\
&\quad 32y_2y_3 + 6y_2y_4 + 14y_2y_5 + 13y_3y_4 + 63y_3y_5 + 87y_4y_5
\end{aligned}$$

Using elementary linear algebra, we finally compute a B^{-1}-like matrix, which takes the v_i to u_iu_{i+1}, and an A^{-1}-like matrix which computes the y_is from the u_is . Both matrices appear in terms of δ and τ.

Once this precomputation has been done, we can forge any signature. For instance, in order to sign the hashed value $(1,2,3,4)$, we randomly choose $v_5 = 44$, and we get, using our equations the valid signature:

$$y_1y_2 = 59\ ,\ y_2y_3 = 60\ ,\ y_3y_4 = 38\ ,\ y_4y_5 = 26\ ,\ y_5y_1 = 56$$

We note that it is perfectly possible to implement the last step "formally" and to sign a formal message $(V_1, V_2, V_3, V_4, V_5)$, thus recovering a substitute to the original signing function.

4 Extensions

4.1 Extension to the case $s > 1$

The case $s > 1$ is more complicated and we only sketch a possible attack. This part has not been implemented. Suppose again that we have k variables $y_1, y_2, \ldots, y_k$, with k odd, whose pairwise products generate the signature, and that the hashed message has $k - s$ quantities $v_1, v_2, \ldots, v_{k-s}$, together with coefficients $c_{ij\ell}$ expressing v_i in terms of $y_j y_\ell$. Suppose for simplicity that $s > 1$ is odd, so that $k - s$ is even.

Some linear combinations of the $k - s$ quadratic forms v_i will have rank $s+1$. Namely, for each index set $I \subseteq \{1, 2, \ldots, k\}$ of size $(s+1)/2$ such that $\forall i, j \in I$: $\mid i - j \mid \geq 2$ and $\{1, k\} \not\subseteq I$, there is such a linear combination of the form

$$\sum_{i \in I} u_i(\alpha_{iI} u_{i-1} + \beta_{iI} u_{i+1}) \tag{24}$$

The number of such index sets I is

$$\frac{k}{\frac{s+1}{2}} \begin{pmatrix} k - \frac{s+3}{2} \\ \frac{s-1}{2} \end{pmatrix} \tag{25}$$

There are more than k linear combinations, leading to increased complication. The space Y_I, spanned by rows of the corresponding quadratic form, contains u_i for each index $i \in I$. So each u_i is in the intersection of a large number of subspaces Y_I, and hopefully only multiples of u_i will be in such an intersection. This algebraic condition should distinguish the u_i, hopefully indexing them by the roots δ of some polynomial $F(\delta)$ of degree k. Pairs $\{u_i, u_{i+2}\}$ of solutions with index differing by 2 should be distinguished by appearing together in many different subspaces Y_I. From this we would be able to distinguish pairs $\{u_i, u_{i+1}\}$. We would fabricate the missing equations as follows: for $j = k - s + 1, \ldots, k$, let $u'_{i(j)}$ be a multiple of u_i, normalized to have a 1 in position j, and set $v'_j = \sum_i u'_{i(j)} u'_{i+1(j)}$.

4.2 The case k=3, s=1

In the special case $k = 3$, $s = 1$, where we must satisfy two quadratic equations in three variables, we can employ an *ad hoc* method, since the methods outlined in section 3 don't work. We take a linear transformation of the two quadratic equations so that the right-hand side of one equation vanishes; that is, if the given values are v_1 and v_2, we take v_2 times the first equation minus v_1 times the second. This gives a homogeneous quadratic equation in three variables y_1, y_2, y_3:

$$\sum_{ij} c_{ij} y_i y_j = 0 \tag{26}$$

The second equation is inhomogeneous:

$$\sum_{ij} d_{ij} y_i y_j = d_0 \tag{27}$$

By setting $z_1 = y_1/y_3$, $z_2 = y_2/y_3$ in (26), we obtain an inhomogeneous quadratic equation in two variables z_1, z_2. We can easily find an affine change of basis from z_1, z_2 to z_1', z_2' which transforms the equation to the form

$$c_{11}' {z_1'}^2 + c_{12}' z_1' z_2' + c_{22}' {z_2'}^2 = c_0' \bmod N \tag{28}$$

and a further linear change of variables to z_1'', z_2'' yielding

$$c_{11}'' z_1''^2 + c_{22}'' z_2''^2 = c_0'' \bmod N \tag{29}$$

which can be solved by the Pollard [7] attack on the Ong-Schnorr-Shamir [5] scheme. We find from this a set of ratios y_j/y_3, and, by extension, a set of ratios $y_i y_j / y_3^2$, satisfying (26). Setting $y_3^2 = \lambda$, the second equation (27) becomes a linear equation in λ. Thus we find a consistent set of pairwise products $y_i y_j$ satisfying the desired equations (26), (27).

4.3 Open questions

The birational permutation signature scheme has many instances, of which we have attacked only the first few examples. For a more complex instance of the scheme, the ideas of the present paper will still apply: the trap door conditions lead to algebraic equations on the coefficients of the transformations, and we hope to gather enough such equations to make it possible to solve them by g.c.d. or Gröbner basis methods. But, for any specific instance, it remains to see whether the ideas of the present paper would be sufficient to mount an attack.

One general theme is that when solutions of the algebraic equations enjoy a symmetry, it makes the equations harder to solve, but we don't need to solve them, since the final solution will enjoy the same symmetry, and quantities symmetric in the roots of the equation can be expressed in terms of the coefficients of the equation alone, not in terms of the roots. When the roots fail to enjoy a symmetry, they can be distinguished by algebraic conditions, which yield further algebraic equations, and the Gröbner basis methods have more to work with. This gives us hope that the methods outlined in this paper will apply with some generality to many instances of the birational permutation signature scheme.

Conclusion

We have shown how to use algorithmic tools to break many of the cryptographic schemes based on birational permutations with hidden trap-doors. Though not all the cases proposed by Shamir have been studied, we demonstrated that use of Galois-like theory may break them. This enlightens cryptanalysis with a new approach. We would like to comment briefly on the mathematics of our attacks. We used pseudo-extensions of pseudo-fields to break the most significant proposals. In a way, this is very similar to the security analysis of RSA-like cryptosystems: stated in a provocative way, this security corresponds to the freedom of treating (at least algorithmically) $\mathbb{Z}_N$ as a field since we *do not know* a non-trivial factor of N. In a similar vein, we took the freedom to consider $\mathbb{Z}_N[\delta]/F(\delta)$ as a Galois extension since we *did not know* how to get a root of F. This is a way to use formally incorrect statements of mathematics in order to achieve actual results.

References

[1] S.R. Czapor, K.O. Geddes and G. Labahn. *Algorithms for Computer Algebra*, Kluwer Academic Press, 1992.

[2] D. Coppersmith, J. Stern and S. Vaudenay. Attacks on the Birational Permutation Signature Schemes. In *Advances in Cryptology CRYPTO'93*, Santa Barbara, California, USA, Lectures Notes in Computer Science 773, pp. 587–593, Springer-Verlag, 1994.

[3] S. Lang. *Algebra.* Second Editon. Addison-Wesley, 1984.

[4] T. Matsumoto and H. Imai. Public Quadratic Polynomial-tuples for Efficient Signature-Verification and Message-Encryption. In *Advances in Cryptology EUROCRYPT'88*, Davos, Switzerland, Lectures Notes in Computer Science 330, pp. 419–453, Springer-Verlag, 1988.

[5] H. Ong, C. P. Schnorr and A. Shamir. A fast signature scheme based on quadratic equations. Proc. 16th ACM Symp. Theory of Computing, pp. 208–216, 1984.

[6] J. Patarin. Cryptanalysis of the Matsumoto and Imai Public Key Scheme of Eurocrypt '88. In *Advances in Cryptology CRYPTO'95*, Santa Barbara, California, USA, Lectures Notes in Computer Science 963, pp. 248–261, Springer-Verlag, 1995.

[7] J. M. Pollard and C. P. Schnorr. An efficient solution of the congruence $x^2 + ky^2 = m \pmod{n}$. IEEE Trans. Inform. Theory vol IT-33 no 5, pp. 702–709, 1987.

[8] R. L. Rivest, A. Shamir and L. M. Adleman. A Method for Obtaining Digital Signatures and Public-key Cryptosystem. In *Communications of the ACM*, vol. 21, pp. 120–126, 1978.

[9] A. Shamir. Efficient signature schemes based on birational permutations. In *Advances in Cryptology CRYPTO'93*, Santa Barbara, California, USA, Lectures Notes in Computer Science 773, pp. 1–12, Springer-Verlag, 1994.

[10] T. Theobald. How to Break Shamir's Asymmetric Basis. In *Advances in Cryptology CRYPTO'95*, Santa Barbara, California, USA, Lectures Notes in Computer Science 963, pp. 136–147, Springer-Verlag, 1995.