Disguising tori and elliptic curves

Steven Galbraith

Mathematics Department, Royal Holloway University of London, Egham, Surrey
TW20 0EX, U.K.

Abstract. Frey proposed the idea of 'disguising' an elliptic curve. This is a method to obtain a 'black box' representation of a group. We adapt this notion to finite fields and tori and study the question of whether such systems are secure.

Our main result is an algebraic attack which shows that it is not secure to disguise the torus T_2. We also show that some methods for disguising an elliptic curve are not secure. Finally, we present a method to disguise an elliptic curve which seems to resist our algebraic attack.

1 Introduction

Frey [4] proposed the idea of 'disguising' an elliptic curve by taking a Weil descent to obtain a group law represented by a system of non-linear multivariate polynomial equations and then 'blinding' this group law by applying an invertible linear change of variable. Frey's proposal was actually to disguise the trace zero part of the Weil restriction, but the basic idea can be applied more generally.

The motivation for such a proposal is to obtain a 'black box' representation for a group. A black box group is a group description for which one can perform the group operation, and possibly test for equality of group elements, but one does not necessarily know the order of the group or anything about the natural representation of group elements. This usually means that the class of algorithms which can be performed in the group is restricted compared to more natural representations of the group. For example, there are index calculus algorithms for finite fields which exploit the 'smoothness' of representations of certain group elements, and these algorithms cannot be implemented in a black box finite field.

This idea has been adapted by Dent and Galbraith [2] for a 'trapdoor pairing' application. It is therefore important to understand the security of such systems.

One of the goals of this work is to study a simplified situation, namely algebraic tori, in the hope is that this will shed light on the elliptic case. Indeed, in Section 3 we obtain methods which show that it is not secure to disguise a torus. We then, in Section 4, explain why some of these attacks do not seem to apply in the elliptic curve case. Further analysis in the elliptic curve case will be required before we can have confidence in its security.

A related computational problem is the isomorphism of polynomials with one secret problem and several algorithms have been proposed to solve this problem. We discuss this work in Section 5.

2 Warm-up: Disguising finite fields

The simplest case is finite fields. We describe this case in some detail as it gives an easy example of the approaches used in the paper, both for disguising a group and also for attacking the resulting system.

Consider a finite field $K = \mathbb{F}_{q^m}$. One can represent elements of K as m-tuples with respect to some basis over $\mathbb{F}_q$ (for example, a polynomial basis). We will use underlining to denote m-tuples, so that the field element a is represented as the m-tuple $\underline{a}$.

There is an explicit way to multiply elements represented as m-tuples. Indeed, there are m homogeneous quadratic polynomials $F_i(x_0, \ldots, x_{m-1}, y_0, \ldots, y_{m-1})$ $(i = 0, \ldots, m-1)$ defined over $\mathbb{F}_q$ such that, if $\underline{a} = (a_0, \ldots, a_{m-1})$ and $\underline{b} = (b_0, \ldots, b_{m-1})$ are m-tuples corresponding to elements $a, b \in K$ then the product $c = ab$ is represented by the m-tuple $\underline{c} = (c_0, \ldots, c_{m-1})$ where

$$c_i = F_i(a_0, \ldots, a_{m-1}, b_0, \ldots, b_{m-1}).$$

We call this a 'natural' representation for $\mathbb{F}_{q^n}$.

We are interested in whether the field K can be diguised to give a black box representation.

The approach taken by Frey for elliptic curves is to apply an invertible transformation U on the vector space $\mathbb{F}_q^m$. Initially, we will assume that U is linear (see the end of this section for a discussion of the more general case). Hence, for any element $\underline{a} = (a_0, \ldots, a_{m-1})$ we associate the 'disguised' element $\underline{a}' = (a'_0, \ldots, a'_{m-1}) = U(a_0, \ldots, a_{m-1})$. One may think of $U(\underline{a})$ as being $M\underline{a}^T$ where M is an $m \times m$ matrix over $\mathbb{F}_q$ and where $\underline{a}^T$ is a column vector (the transpose of $\underline{a}$).

Denote by $\underline{x}'$ and $\underline{y}'$ two m-tuples representing arbitrary disguised group elements. To obtain a group law it is necessary to 'disguise' the polynomials F_i describing the group law to get m polynomials

$$(F'_0, \ldots, F'_{m-1}) = U\left(F_0(U^{-1}\underline{x}', U^{-1}\underline{y}'), \ldots, F_{m-1}(U^{-1}\underline{x}', U^{-1}\underline{y}')\right).$$

defined over $\mathbb{F}_q$ (since U is linear the F'_i are still quadratic). One can easily check that if $\underline{a}', \underline{b}'$ are disguised representations of $a, b \in K$ then $\underline{c}' = (F'_i(\underline{a}', \underline{b}'))$ is a disguised representation of $ab \in K$. Since U is linear, the addition operation in the field can also be immediately computed.

It is trivial to see that this 'disguised' representation does not give a black box group. We stress that the issue is not whether one can find the transformation U, but whether one can find a way to interpret the 'disguised' representation as a 'natural' representation (i.e., to 'look inside the box'). Hence the solution to the cryptanalysis problem is not unique.

One attack is to recover a polynomial representation of the field using the following method. Choose a random m-tuple $\underline{w}'$ corresponding to a field element w and use the group operation to compute blinded representations of $w^2, w^3, \ldots,$ w^m, w^{m+1}. One then has $m+1$ vectors in $\mathbb{F}_q^m$ so there is a linear dependence over $\mathbb{F}_q$. This linear dependence gives a polynomial $g(x)$ which (once the trivial factor x is removed) has degree m over $\mathbb{F}_q$ and has w as a root. If w does not lie in a proper subfield then $g(x)$ is irreducible and we have recovered a natural polynomial representation of our 'disguised' finite field K. It is easy to decompose other elements of the disguised group representation in terms of the new polynomial basis (indeed, computing isomorphisms between finite fields is easy [7]). One can then apply index calculus algorithms etc.

We assumed above that U is linear. Frey [4] actually assumed affine maps, so that $U(\underline{x}) = M\underline{x}^T + \underline{t}^T$ for some matrix M and vector $\underline{t}$. There are at least two ways to show that there is no extra generality by doing this. One approach (following Perret [11]) is to note that $M\underline{x}^T + \underline{t}^T$ can be expressed as $M'(\underline{x}')^T$ where $\underline{x}' = (\underline{x}, 1)$ and

$$M' = \begin{pmatrix} M & \underline{t}^T \\ \underline{0} & 1 \end{pmatrix}.$$

Another approach is to note that the equation $0 \cdot 0 = 0$ in K implies that the disguised representation of 0, namely $\underline{t}$, is a fixed point of the multiplication operation (another fixed point comes from $1 \cdot 1 = 1$). One can find such fixed points from the equation $(F_i'(\underline{x}, \underline{x})) = \underline{x}$, which we will assume can be easily solved using Gröbner basis methods. Once $\underline{t}$ is found one can perform a transformation to reduce the problem to the previous case.

The case where U is non-affine is a question for further research, but such transformations are less interesting from the application point of view as they would increase the degree of the equations defining the group operation.

To summarise, it is impossible to securely disguise a finite field in the above manner. We remark that our discussion does not contradict the results of Boneh and Lipton [1] since the 'disguised' field above is not a true black box field: the subfield $\mathbb{F}_q$ is clearly visible and the addition operation is not 'black box'.

3 Tori

We now consider algebraic tori. These are subgroups of the multiplicative group of a finite field.

Consider the simplest non-trivial torus, namely the subgroup $T_2 \subset \mathbb{F}_{q^{2m}}^*$ of order $q^m + 1$. Equivalently, T_2 is the kernel of the norm map with respect to the Galois extension $\mathbb{F}_{q^{2m}}/\mathbb{F}_{q^m}$. Assume that $\mathbb{F}_{q^{2m}} = \mathbb{F}_{q^m}(\alpha)$ for some element α. We denote by $\overline{\alpha}$ the Galois conjugate of α.

There are several ways to represent this torus. The 'direct' way is $T_2 = \{a + b\alpha : (a + b\alpha)(a + b\overline{\alpha}) = 1\}$ which allows one to represent T_2 as an affine variety in $\mathbb{F}_{q^m}^2$. Another commonly used way to represent this tori is in the 'affine representation'

$$T_2 \backslash \{1\} = \left\{ \frac{a + \alpha}{a + \overline{\alpha}} : a \in \mathbb{F}_{q^m} \right\}.$$

A third way to represent the torus is in the 'projective representation'

$$T_2 = \left\{ \frac{a + b\alpha}{a + b\overline{\alpha}} : a, b \in \mathbb{F}_{q^m}, (a, b) \neq (0, 0) \right\}$$

which has a natural projective equivalence relation such that the element corresponding to (a, b) is equivalent to the element corresponding to $(\lambda a, \lambda b)$ for any $\lambda \in \mathbb{F}_{q^m}^*$.

For the affine and projective representations it is sufficient to store and compute with the numerator only (since the denominator is always the Galois conjugate). Hence, the projective representation essentially coincides with the direct representation, except the norm 1 condition on $a + b\alpha$ is no longer required.

There are other algebraic tori used in cryptography, such as the subgroup T_6 of $\mathbb{F}_{q^6}^*$ of size $q^2 - q + 1$. However, the problem of disguising such tori seems to be of less interest since the group law directly on the torus is unattractive in these cases. Instead, in practice one tends to 'decompress' the representation, compute the group operations, and then 'recompress'. An exception is the XTR representation (which is not a torus) and it might be of interest to study disguising this group operation.

3.1 Representing the group law in T_2

For simplicity let us now restrict to the case where m is odd and $q = 2^s$ where s is odd. Then we can assume that $\alpha^2 + \alpha + 1 = 0$ and $\overline{\alpha} = \alpha + 1$. The group law on the direct or projective representations of T_2 is given by

$$(a + b\alpha)(c + d\alpha) = ac + bd + \alpha(ad + bc + bd).$$

We call this the 'projective group law'. For the affine representation this is

$$(a + \alpha)(c + \alpha) = \frac{ac + 1}{a + c + 1} + \alpha.$$

Note that addition of elements of a torus is not defined.

As before, fix a basis for $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$ and represent an element $a \in \mathbb{F}_{q^m}$ by an m-tuple $(a_0, \dots, a_{m-1})$. The above group law can be expressed in terms of m-tuples. The affine multiplication rule is represented by m rational functions in $2m$ variables while the projective multiplication rule is represented by $2m$ quadratic polynomials in $4m$ variables.

The projective group law can be iteratively composed, but the affine representation is not so convenient. Hence, in practice, one would prefer the projective representation. The affine multiplication can be expressed as a system of polynomial equations by giving an inversion rule using the norm. This is convenient in the case $s = 1$, since the norm of any invertible element with respect to $\mathbb{F}_{2^m}/\mathbb{F}_2$ is 1. We omit the details and stick with using the projective formulation.

To summarise, the projective group law is given by a sequence of $2m$ quadratic polynomials $F_i(a, b, c, d)$ in $4m$ variables. In the case where T_2 has the direct representation the polynomials are exactly the same, so this representation is identical to the projective version even though we should 'officially' only work with elements $(a + b\alpha)$ of norm 1.

To disguise the group we apply an invertible transformation U of $2m$-dimensional space. We therefore obtain the blinded affine system

$$F_i' = UF_i\left(U^{-1}(a, b), U^{-1}(c, d)\right)$$

The hope is that this blinded group law forms a black box group. Of course, the public data contains q and $2m$ so we know that the black box group represents the torus of order $q^m + 1$. If the group operations alone are published then one cannot necessarily compute inverses, test equality of projective representations or even recognise a representative of the identity element 1.

3.2 Algebraic attack

Consider first the special case where the transformation U is of the form $U_1 \times U_2$ where U_1 is a transformation of the first m-tuple (representing a) and where U_2 is a transformation of the second m-tuple (representing b). We show that it is trivial to recover the 'natural' structure of the torus as a subgroup of $\mathbb{F}_{q^{2m}}^*$ in this case.

The attack begins by taking a random element w of the form $(a, 0)$ by setting the second half of the values in the $2m$-tuple to be zero. The multiplication rule on elements of the form $(a, 0) = a + 0\alpha$ becomes simply the multiplication rule in $\mathbb{F}_{q^m}$. Hence one computes $(a^2, 0), (a^3, 0), \ldots, (a^{m+1}, 0)$ and recovers a polynomial representation for $\mathbb{F}_{q^m}$ as done in the previous section.

One then takes a random element of the form $u = (0, \star)$. Then u corresponds to $b\alpha$ for some value b. One can compute $u^2 = b^2\alpha^2$ in the form (a', b'). We have $a' = b^2$ and $b' = b^2$. One can express a' with respect to the newly obtained polynomial basis for $\mathbb{F}_{q^m}$ and hence recover the representation of b with respect to this basis. Repeating for m judicious choices of u allows all $2m$-tuples in the disguised representation to be expressed in the form $a + b\alpha$ where a and b are expressed in terms of a polynomial basis for $\mathbb{F}_{q^m}$. The natural structure of the torus is therefore recovered.

We remark, for later reference, that the key to this attack is using the operation $1 \cdot 1 = 1$ on varying representations of 1 to turn our black box group operation into a black box for multiplication in a finite field.

We now consider the more general case, where the transformation U mixes a and b variables together. The attack is similar. Choose a random $2m$-tuple (a, b) and use the group operation to compute $w = (a', b') = (a, b)^{q^m+1}$. Then (a', b') is an element which represents the identity, and so it corresponds to an un-disguised element of the form $(\star, 0)$. Once we have such a random element we can recover a polynomial basis for $\mathbb{F}_{q^m}$ as above. One can then apply a linear transformation U_1 which diagonalises the first m variables. Again, choose a random element of the form $u = (0, \star)$. In this case, all we know is that $u = a + b\alpha$ for some $a, b \in \mathbb{F}_{q^m}$. Compute $wu, \ldots, w^m u$, which are all of the form $a' + b'\alpha$. One can then find a linear transformation U_2 which removes all the terms a, a' etc, so that the original value $u = (0, \star)$ is now 'purely quadratic'. A natural representation of the torus is now recovered.

Example: We take $m = 3$ and consider the torus in $\mathbb{F}_{2^6}^*$ of order $2^3 + 1$. We represent $a \in \mathbb{F}_{2^3}$ as $(a0, a1, a2)$.

The un-disguised group law to multiply $(a + b\alpha)(c + d\alpha)$ is given by the 6 polynomials (thanks to Magma):

$$\begin{aligned}
F_0 &= a0c0 + a1c2 + a2c1 + b0d0 + b1d2 + b2d1\\
F_1 &= a0c1 + a1c0 + a1c2 + a2c1 + a2c2 + b0d1 + b1d0 + b1d2 + b2d1 + b2d2\\
F_2 &= a0c2 + a1c1 + a2c0 + a2c2 + b0d2 + b1d1 + b2d0 + b2d2\\
F_3 &= a0d0 + a1d2 + a2d1 + b0c0 + b0d0 + b1c2 + b1d2 + b2c1 + b2d1\\
F_4 &= a0d1 + a1d0 + a1d2 + a2d1 + a2d2 + b0c1 + b0d1 + b1c0 + b1c2\\
&\quad + b1d0 + b1d2 + b2c1 + b2c2 + b2d1 + b2d2\\
F_5 &= a0d2 + a1d1 + a2d0 + a2d2 + b0c2 + b0d2 + b1c1 + b1d1 + b2c0\\
&\quad + b2c2 + b2d0 + b2d2
\end{aligned}$$

Now apply a linear change of variable U to get the blinded system:

$$\begin{aligned}
F0 &= a0c0 + a0c2 + a0d1 + a1d0 + a1d2 + a2c0 + b0c1 + b1c0\\
&\quad + b1d1 + b1d2 + b2c1 + b2d1\\
F1 &= a0c0 + a0d0 + a0d1 + a1c1 + a1c2 + a1d0 + a2c1 + b0c0\\
&\quad + b0c1 + b0d2 + b1c0 + b1d1 + b2d0 + b2d2\\
F2 &= a0c1 + a0d1 + a1c0 + a1c1 + a1d1 + a2c2 + b0d1 + b1c0\\
&\quad + b1c1 + b1d0 + b1d1 + b1d2 + b2d1 + b2d2\\
F3 &= a0c1 + a1c0 + a1c1 + a1d0 + a2d0 + b0c1 + b0c2 + b0d1\\
&\quad + b1d0 + b1d2 + b2d1\\
F4 &= a0d0 + a0d2 + a1c1 + a1d0 + a1d2 + a2d1 + b0c0 + b0c1\\
&\quad + b0d0 + b1c2 + b1d2 + b2c0 + b2c1 + b2d1\\
F5 &= a0c0 + a0c1 + a0d1 + a1c0 + a1c1 + a2d2 + b0d0 + b0d1\\
&\quad + b1c0 + b1d0 + b1d2 + b2c2 + b2d1 + b2d2.
\end{aligned}$$

Choose $u = (a0, a1, a2, b0, b1, b2) = (1, 0, 0, 0, 0, 0)$. One computes $w = u^9 = u^{2^3+1} = (1, 1, 0, 0, 1, 0)$. This represents an element in $\mathbb{F}_{2^3}$ corresponding to the torus element 1. Now, $w^2 = (0, 1, 0, 1, 1, 0)$, $w^3 = (1, 1, 1, 0, 1, 0)$ and $w^4 = (1, 0, 0, 1, 0, 0)$. One finds the linear relation $w + w^2 + w^4 = (0, 0, 0, 0, 0, 0)$. Hence the element w has minimal polynomial $g(x) = x^3 + x + 1$. Acting by the matrix

$$U_1 = \begin{pmatrix} 0&0&1&0&0&0\\ 1&0&0&0&0&0\\ 0&0&0&1&0&0\\ 0&1&0&0&1&0\\ 1&0&0&1&1&0\\ 0&0&0&0&0&1 \end{pmatrix}$$

transforms the representation so that the first three components correspond to $\mathbb{F}_{2^3}$ with the polynomial basis $\{1, w, w^2\}$. The polynomials F_i should be transformed under U_1.

Now take $u = (0, 0, 0, 1, 0, 0)$. One finds that $1u = u$, $wu = (1, 0, 0, 0, 1, 0)$ and $w^2u = (0, 1, 0, 0, 1, 1)$. Hence, u does not correspond to a purely quadratic field element such as $b\alpha$. Transforming under a matrix of the form

$$U_2 = \begin{pmatrix} I_3 & A \\ 0 & I_3 \end{pmatrix}$$

where I_3 is a 3×3 identity matrix and

$$A = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

means that the element $u = (0, 0, 0, 1, 0, 0)$ is now purely quadratic. Calling it α we can compute $w\alpha = (0, 0, 0, 1, 0)$ and $w^2\alpha = (0, 0, 0, 0, 1, 1)$. Hence we now have a completely explicit basis in terms of the elements w and α. Finally, one computes $\alpha^2 = (0, 1, 1, 0, 0, 1)$ and finds that α satisfies the equation $\alpha^2 + (w + w^2)(\alpha + 1) = 0$. A natural representation for the torus is therefore obtained. If required, one could compute an isomorphism to a 'nicer' basis using the methods of Lenstra [7].

4 Disguising elliptic curves

We now consider the case of the most interest, which is to obtain a black box representation of an elliptic curve group. The minimum requirement for a black box group is to be able to compute the group operation and to be given an element of the group. Features which may or may not be given include the ability to test equality of group elements, knowledge of the order of the group, and the ability to randomly sample elements of the group

4.1 Elliptic curve group operations

As with tori, the most practical approach is to work with projective equations. Hence, assume we have a curve

$$y^2z + a_1xyz + a_3yz^2 = x^3 + a_2x^2z + a_4xz^2 + a_6z^3$$

over a finite field $\mathbb{F}_{q^m}$. As usual, we will represent elements $x \in \mathbb{F}_{q^m}$ as m-tuples over $\mathbb{F}_q$.

We want to describe the group operation $(x : y : z) + (u : v : w)$ as polynomials. We will require $3m$ polynomials in $6m$ variables.

Since the attack on tori exploits the operation $1 \cdot 1 = 1$, it is natural in the elliptic curve case to exploit the operation $0 + 0 = 0$ (where 0 is the point at infinity on the curve). It turns out (see [5, 6]) that it is impossible to give one list of polynomials for elliptic curve addition which give the correct result for all valid input points. This fact seems to thwart the algebraic attacks used earlier.

The natural thing to do is to use the usual group law for affine points (i.e., where $z \neq 0$) which is extended to projective points. There are formulae for doubling and formulae for addition, or one might prefer the combined formulae which represent both doubling and addition. It is sufficient for the attack on tori to consider squaring, so we focus on doubling here.

For a curve of the form $y^2 = x^3 + Ax + B$ the usual formula is

$$[2](x, y, z) = (x2, y2, z2)$$

where

$$\begin{aligned} x2 &= 2yz(x^4 - 2Ax^2z^2 - 8Bxz^3 + A^2z^4) \\ y2 &= x^6 + 5Ax^4z^2 + 20Bx^3z^3 - 5A^2x^2z^4 - 4ABxz^5 - (A^3 + 8B^2)z^6 \\ z2 &= (2yz)^3. \end{aligned}$$

One can see that $[2](0, 1, 0) = (0, 0, 0)$ so this formula is not valid when doubling infinity.

The addition formulae are considerably more complicated, and not just because there are 6 variables involved instead of 3. We refer to [2] for some details.

A case of particular interest for the hidden pairing application is disguising supersingular elliptic curves. So consider $y^2 + y = x^3 + a_2x^2 + a_4x + a_6$. The projective doubling formula in this case is $[2](x, y, z) = (x2, y2, z2)$ where

$$\begin{aligned} x2 &= x^4z^2 + (a_4^2 + a_2)z^6 \\ y2 &= x^6 + a_4x^4z^2 + x^3z^3 + (a_4^2 + a_2)x^2z^4 + a_4xz^5 + yz^5 + (a_2a_4 + a_4^3 + 1)z^6 \\ z2 &= z^6. \end{aligned}$$

Again, one sees that $[2](0, 1, 0)$ gives $(0, 0, 0)$, which is not defined.

4.2 Disguising an elliptic curve

Let $F_i(\underline{x}, \underline{u})$ be a system of polynomial functions which represents the elliptic curve group law. This may be a general group addition formula, or one could use two systems of polynomial functions, the first giving point doubling and the second giving addition of two distinct points.

As before, we take an invertible transformation U which acts on $3m$-tuples and define the 'disguised' $3m$-tuples as $(\underline{x}', \underline{y}', \underline{z}') = U(\underline{x}, \underline{y}, \underline{z})$. The 'disguised' group law is obtained as

$$\left(F_i'(\underline{x}', \underline{y}', \underline{z}', \underline{u}', \underline{v}', \underline{w}')\right) = U\left(F_i(U^{-1}(\underline{x}', \underline{y}', \underline{z}'), U^{-1}(\underline{u}', \underline{v}', \underline{w}')\right)$$

4.3 Attacks

As before we consider two cases.

In the first case, suppose U keeps the m-tuples corresponding to x, y and z separate. The natural first guess would be to set all x and z variables to zero and to compute $(0, y, 0) + (0, y, 0)$. But this is not useful since the group operation is not defined on such points (it returns $(0, 0, 0)$). This is a crucial observation.

Nevertheless, in this case one can set, say, $y = z = 0$ and compute $[2](x, 0, 0) = (0, x^6, 0)$. If U acts the same way on each m-tuple then one could then compute $[2](x^6, 0, 0) = (0, x^{36}, 0)$ and recover x as long as the extension degree is coprime to 6. Note that it doesn't matter if $(x, 0, 0)$ is a point on the curve or not; we are simply evaluating some polynomials on some values.

Now consider the case where U mixes all variables together. In this case the algebraic attacks do not seem to work. Even if the group order N is known, one can compute $[N](x, y, z)$ for a random point, to get a representation of a point $(0, 1, 0)$. But, as noted, the group law fails to add points at infinity correctly and so the black box elliptic curve group law does not seem to give us access to a black box finite field.

Hence, elliptic curves seem to resist the algebraic methods used to attack the case of the torus T_2.

On the other hand, they might be attacked by multivariate techniques, such as linearisation or Gröbner bases, which would determine the transformation U just from the polynomials defining the group operation. We discuss such attacks in more details in the next section.

5 Relationship with the isomorphism of polynomials problem

Some related problems have been already considered in the literature. Let $\underline{a} = (a_1(x_0, \dots, x_{m-1}), \dots, a_n(x_0, \dots, x_{m-1}))$ be a list of n polynomials in m variables over $\mathbb{F}_q$. Let $\underline{b}$ be another such list of n polynomials in m variables.

Isomorphism of polynomials with one secret problem (IP1S): Given $\underline{a}$ and $\underline{b}$ find $M \in \mathrm{GL}_m(\mathbb{F}_q)$ and $\underline{t}$ such that $b_i(x) = a_i(M\underline{x}^T + \underline{t}^T)$.

Isomorphism of polynomials (with two secrets) problem (IP): Given $\underline{a}$ and $\underline{b}$ find $M, N \in \mathrm{GL}_m(\mathbb{F}_q)$ and $\underline{t}_1, \underline{t}_2$ such that $b_i(x) = Na_i(M\underline{x}^T + \underline{t}_1^T) + \underline{t}_2^T$.

Polynomial linear equivalence (PLE): Given $\underline{a}$ and $\underline{b}$ find $M \in \mathrm{GL}_m(\mathbb{F}_q)$ such that $b_i(x) = a_i(M\underline{x}^T)$.

The isomorphism of polynomials problem was introduced by Patarin [10]. Perret [11] shows that IP1S and PLE are equivalent, by using the fact that an affine transformation on n variables can be viewed as a linear transformation on $n + 1$ homogeneous variables.

Perret [11] gives an interesting algorithm to attack the PLE problem, which is based on getting information about M by considering the Jacobian matrices of the systems $\underline{a}$ and $\underline{b}$ (essentially relating the degree 1 components of $\underline{a}$ and $\underline{b}$) and similarly relating the degree 2 components of $\underline{a}$ and $\underline{b}$. This algorithm solves many PLE problems in polynomial time.

More recently, Faugère and Perret [3] consider solving such systems using the F_5 Gröbner basis algorithm. Advantage is taken of working with homogeneous components of low degree, when possible. They give experimental results in a number of cases, and give some idea of the numbers of variables required to prevent their attack. Unfortunately, it seems to be hard to give a complexity estimate for their methods, or even a clear picture of the number of variables required to attain a given security level.

The computational problem arising from our application is not necessarily as hard as the general IP problem, since we blind our system using a single matrix, as $F'(\underline{x}, \underline{u}) = UF(U^{-1}\underline{x}, U^{-1}\underline{u})$.

On the other hand, for the IP1S and PLE problems both the original system of polynomials a and the final system b are given. In our application, only the final system F' is given to an attacker. However, if information about F is leaked or can be guessed then attacks of this form may be relevant. This is particularly relevant for the case of hidden pairings [2], where the original equation of the curve can be guessed.

5.1 Hidden pairings

In [2] it is proposed to disguise the supersingular elliptic curve $y^2 + y = x^3 + 1$ over $\mathbb{F}_{q^m}$ where $q = 2^s$. Two variants are given, the first of which involves publishing polynomials giving a general addition rule for pairs of points in blinded representation. The second variant gives a partial group law, comprising a general doubling formula, but only formulae for addition by a fixed point P. Note that, in both variants, the published polynomial systems for the doubling operation are homogeneous of degree at least 3, though in the second variant the addition formula with respect to a fixed base point P is not homogeneous.

In both variants, one can obtain pairs $(F(\underline{x}), F'(\underline{x}'))$ of systems of explicitly known polynomials such that $F'(\underline{x}') = U(F(U^{-1}\underline{x}'))$ where U is the unknown change of variable, by considering just the case of doubling a point. One can also obtain other systems of polynomials from the group law, but it seems natural to start with the simplest case of doubling. Hence, the security of the scheme depends on an the hardness of a certain isomorphism of polynomials problem.

In [2] a method to solve such systems using Gröbner bases is sketched. Much more detail of this sort of attack is given in [3]. Further work on determining which, if any, values for m are secure is needed.

The reason why this attack is applicable is that it is known how the original curve and group law are chosen. For example, this attack cannot be performed using the polynomial equations representing addition by the fixed point P, since the original coordinates of P cannot be guessed.

One might think that this attack can be avoided by taking a more general elliptic curve equation, as suggested at the send of Section 4.2 of [2]. Unfortunately, this idea does not work, since all supersingular elliptic curves over $\mathbb{F}_{2^n}$ are isomorphic over $\mathbb{F}_{2^n}$ to one of a finite number of 'canonical' supersingular elliptic curves (see, for example, [8, 9]). Since an isomorphism of Weierstrass equations of elliptic curves over $\mathbb{F}_{2^n}$ is given by a linear change of variable, any such isomorphism may be already included in the change of variable U. Hence, if the curve equation is 'randomised' but the isomorphism of polynomials problem can be efficiently solved, then we can just repeat the attack a small number of times by trying each of the canonical elliptic curves in turn.

To summarise, the Gröbner basis methods in [2, 3] do not seem to be directly applicable to the general problem of disguising an elliptic curve, but they are applicable to the hidden pairings application. As a result, for hidden pairings it seems that m is required to be rather large (at least $m \geq 11$) and so the storage requirements for the system might be too large for it to be practical.

6 Conclusions

We have given a simple algebraic attack which shows that it is not secure to disguise the torus T_2. We have then explained why this attack does not seem to apply to disguised elliptic curves. Finally, we have considered the case of hidden pairings and explained the connection with the isomorphism of polynomials problem. Due to the success of the methods in [3] it seems that the parameter m in hidden pairing applications is required to be larger than hoped in [2], and so the practicality of such systems is questionable.

We encourage further research on cryptanalysis of disguised elliptic curve systems.

7 Acknowledgements

We thank Christopher Wolf, Alex Dent and Ludovic Perret for comments. The work described in this paper has been supported in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT.

References

1. D. Boneh and R. Lipton, Algorithms for black box fields and their application to cryptography, in N. Koblitz (ed.), CRYPTO '96, Springer LNCS 1109 (1996) 283–297.
2. A. W. Dent and S. D. Galbraith, Hidden pairings and trapdoor DDH groups, to appear in ANTS-VII (2006).
3. J.-C. Faugère and L. Perret, Polynomial equivalence problems: algorithmic and theoretical aspects, in S. Vaudenay (ed.), EUROCRYPT 2006, Springer LNCS ???? (2006) ??-??.
4. G. Frey, How to disguise an elliptic curve (Weil descent), Talk at ECC 1998. Slides available from:
`http://www.cacr.math.uwaterloo.ca/conferences/1998/ecc98/frey.ps`
5. H. Lange and W. Ruppert, Complete systems of addition laws on abelian varieties, *Invent. Math.*, **79** (1985) 603–610.
6. H. W. Lenstra Jr., Elliptic curves and number-theoretic algorithms, Proceedings of the International Congress of Mathematicians (1986).

7. H. W. Lenstra Jr., Finding isomorphisms between finite fields, *Math. Comp.*, **56**, No. 193 (1991) 329–347.
8. A. J. Menezes and S. Vanstone, Isomorphism classes of elliptic curves over finite fields of characteristic 2, *Utilitas Mathematica*, 38 (1990) 135–154.
9. A. J. Menezes, Elliptic Curve Public Key Cryptosystems Kluwer, 1993.
10. J. Patarin, Hidden fields equations (HFE) and isomorphisms of polynomials (IP): two new families of asymmetric algorithms, in U. M. Maurer (ed.), EUROCRYPT '96, Springer LNCS 1070 (1996) 33–48.
11. L. Perret, A fast cryptanalysis of the isomorphism of polynomials with one secret problem, in R. Cramer (ed.), EUROCRYPT 2005, Springer LNCS 3494 (2005) 354–370.