Aspects of Pairing Inversion

S. Galbraith, and F. Hess, and F. Vercauteren

Abstract—We discuss some applications of the pairing inversion problem and outline some potential approaches for solving it. Our analysis of these approaches gives further evidence that pairing inversion is a hard problem.[1]

Keywords: pairing inversion, Tate pairing, ate pairing, Diffie-Hellman problem

I. Introduction

Pairing-based cryptography is a major area of research in public key cryptography. The security of pairing-based cryptosystems relies on the difficulty of solving various computational problems. Some of these computational problems have only been very recently proposed, and there has been little scrutiny in the literature of whether they are truly difficult.

This paper studies a collection of such computational problems namely, the problems of inverting various pairings on elliptic or hyperelliptic curves. We describe some potential avenues for solving some pairing inversion problems and we discuss the limitations of these approaches.

We present several results on applications of pairing inversion. It is well known (following Verheul) that if one can invert certain pairings on a class of curves then one can solve the computational Diffie-Hellman problem (CDH) in a class of subgroups of finite fields. This shows that the difficulty of pairing inversion problems has implications not just to pairing-based cryptography, but also to all cryptography based on exponentiation in finite fields. Verheul's results [22], [23] (and also those of Satoh [19]) are usually considered as evidence for the difficulty of pairing inversion.

We give some applications of being able to solve certain restricted pairing inversion problems. For example, we show that it is sufficient to solve just a one-sided pairing inversion problem to be able to solve the bilinear-Diffie-Hellman problem.

Pairings on (hyper)elliptic curves are computed in two stages. The first stage is to perform Miller's algorithm (which computes the evaluation of a certain function at a certain divisor). The second stage is the final exponentia-

Mathematics Department, Royal Holloway University of London, Egham, Surrey TW20 0EX, UK. **steven.galbraith@rhul.ac.uk**

Technische Universität Berlin, Fakultät II, Institut für Mathematik Sekr. MA 8-1, Strasse des 17. Juni 136, D-10623 Berlin, Germany. **hess@math.tu-berlin.de**

Department of Electrical Engineering, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium. **frederik.vercauteren@esat.kuleuven.be**

tion (typically, exponentiation in $\mathbb{F}_{q^k}^*$ to a power $(q^k-1)/N$ where N is to be defined later). Hence, naively, to invert a pairing seems to require first inverting the final exponentiation and then inverting Miller's algorithm.

It was shown by Galbraith, Ó hÉigeartaigh and Sheedy [11] and Granger et al. [12] that some pairings can be computed without a final exponentiation (or with final exponentiation reduced to just a squaring). Hence, in these cases, the difficulty of pairing inversion depends entirely on the difficulty of inverting Miller's algorithm.

On the other hand, we discuss in Section VI cases where inverting Miller's algorithm is easy. However, in these cases the final exponentiation is highly non-trivial (and is many-to-one). Hence, in these cases, the difficulty of pairing inversion is entirely due to the difficulty of finding the right pre-image of the final exponentiation.

One might then conclude that if either finding the right pre-image of the final exponentiation or inverting Miller's algorithm is hard, inverting the pairing is hard. However, it might be possible to invert pairings in one step (rather than the two stage process mentioned above). In Section VII we discuss approaches along these lines, and give the rather subtle reasons why they do not seem to work.

The plan of the paper is as follows. In Sections II and III, we define the pairing inversion problems and give several applications. After reviewing the existing pairings in Section IV, we restate the pairing inversion problems in Section V. In Sections VI and VII we consider approaches to solve pairing inversion problems in the elliptic curve case. To end, we consider curves of higher genus in Section VIII.

II. Statement of the problems

The main technical contents of the paper concern the ate pairing, which is defined on the product of two distinct cyclic subgroups of elliptic curves (or divisor class groups of higher genus curves). Hence, we define our pairings on cyclic groups.

Let G_1, G_2 and G_T be cyclic groups of prime order r. In this paper we consider non-degenerate bilinear pairings of the form

$$e : G_1 \times G_2 \longrightarrow G_T.$$

We now define the two pairing inversion problems under consideration.

Definition 1: Let e be a non-degenerate bilinear pairing as above.

The **Fixed Argument Pairing Inversion 1 (FAPI-1)** problem is: Given $\overline{D}_1 \in G_1$ and $z \in G_T$, compute $\overline{D}_2 \in G_2$ such that $e(\overline{D}_1, \overline{D}_2) = z$.

The **Fixed Argument Pairing Inversion 2 (FAPI-2)** problem is: Given $\overline{D}_2 \in G_2$ and $z \in G_T$, compute $\overline{D}_1 \in G_1$ such that $e(\overline{D}_1, \overline{D}_2) = z$.

Note that both problems FAPI-i have a unique solution for each given pair $(\overline{D}_i, z) \in G_i \times G_T$, as long as $\overline{D}_i \neq 0$, since the pairing is non-degenerate and the groups G_1, G_2 and G_T are cyclic of order r.

We remark that one can solve the discrete logarithm problem in G_T in subexponential time (in terms of the input size of G_T) and hence one can solve FAPI-1 and FAPI-2 in subexponential time (in terms of the size of the input (D, z)). Our concern in this paper is whether one can do better than this, in particular whether there are families of groups for which pairing inversion is polynomial time. Note that one can solve the discrete logarithm problem in G_1 or G_T using Pollard's methods but this has exponential complexity for families with bounded embedding degree.

Finally, we mention a more general problem.

Generalised Pairing Inversion (GPI): Given a pairing e and a value $z \in G_T$, find $\overline{D}_1 \in G_1$ and $\overline{D}_2 \in G_2$ with $e(\overline{D}_1, \overline{D}_2) = z$.

Obviously, GPI is not harder than either FAPI-1 or FAPI-2.

III. APPLICATIONS OF PAIRING INVERSION

In this section we explore some applications of the ability to solve pairing inversion problems. Note that for fixed groups G_1, G_2 and G_T it is often the case that there are several alternative ways to define or implement pairings $e : G_1 \times G_2 \to G_T$. For most of the following applications it will be sufficient to be able to invert any one of the different pairings.

We start by generalising the result of Verheul to the case of pairings on cyclic groups $G_1 \neq G_2$.

Theorem 2: Let $e : G_1 \times G_2 \to G_T$ be a non-degenerate bilinear pairing on cyclic groups of prime order r. Suppose one can solve FAPI-1 and FAPI-2 in polynomial time. Then one can solve the computational Diffie-Hellman problem (CDH) in G_1, G_2 and G_T in polynomial time.
PROOF: Let O_1 be an oracle to solve FAPI-1 for e and let O_2 be an oracle to solve FAPI-2 for e. In other words, $O_1(P, z)$ returns $Q \in G_2$ such that $e(P, Q) = z$.

Let (P, aP, bP) be a CDH input in G_1. Choose a random $Q \in G_2$ and compute $z = e(aP, Q)$. Call $O_1(P, z)$ to get aQ. Now compute $z' = e(bP, aQ)$ and call $O_2(Q, z')$ to get abP.

The other two cases are similar. $\square$

This shows that if one can efficiently solve both pairing inversion problems FAPI-1 and FAPI-2 for a family of curves then the corresponding subgroups G_T in finite fields are not secure for cryptography.

Now we present new results.

Lemma 3: Let notation be as above. If one can solve FAPI-1 in polynomial time then one can compute all non-trivial group homomorphisms $\psi_1 : G_1 \to G_2$ in polynomial time.
PROOF: Fix generators $P \in G_1$ and $Q \in G_2$. We will show how to compute the unique group homomorphism $\psi_1 : G_1 \to G_2$ defined by $\psi_1(P) = Q$. Since any non-trivial group homomorphism maps a generator onto a generator, we will be able to compute all non-trivial group homomorphisms in this way.

Let O_1 be an oracle to solve FAPI-1. Given any $P' \in G_1$ we know that $P' = aP$ for some $a \in \mathbb{Z}/r\mathbb{Z}$. Compute $z = e(P', Q) = e(P, Q)^a$. Call $O_1(P, z)$ to get $aQ = \psi_1(P')$ as required. $\square$

Similarly, one can prove.

Lemma 4: Let notation be as above. If one can solve FAPI-2 in polynomial time then one can compute all non-trivial group homomorphisms $\psi_2 : G_2 \to G_1$ in polynomial time.

Hence, inverting pairings enables the computation of "distortion maps" between G_1 and G_2. We will now present some applications of this idea.

Corollary 5: If one can solve FAPI-1 in polynomial time then one can solve DDH in G_1 in polynomial time. If one can solve FAPI-2 in polynomial time then one can solve DDH in G_2 in polynomial time.
PROOF: Let P, aP, bP, cP be a DDH problem instance in G_1. Let $Q \in G_2$, then by Lemma 3 one can compute a homomorphism $\psi_1 : G_1 \to G_2$ such that $\psi_1(P) = Q$. So compute $aQ = \psi_1(aP)$ and $cQ = \psi_1(cP)$ and test whether $e(P, cQ) = e(bP, aQ)$ as usual. The second statement follows analogously. $\square$

Theorem 6: Let notation be as above. Then the following are equivalent
1. One can solve FAPI-1 and FAPI-2 in polynomial time;
2. One can solve FAPI-1 in polynomial time and one has an efficiently computable (i.e., computable in polynomial time) homomorphism $\psi_2 : G_2 \to G_1$;
3. One can solve FAPI-2 in polynomial time and one has an efficiently computable homomorphism $\psi_1 : G_1 \to G_2$.

PROOF: $(1) \Rightarrow (2)$ is Lemma 4 and $(1) \Rightarrow (3)$ is Lemma 3.

We now show $(2) \Rightarrow (1)$. Given a FAPI-2 instance (Q, z) set $P = \psi_2(Q)$ and run the FAPI-1 oracle $O_1(P, z)$ to get Q'. Then $Q' = aQ$ for some $a \in \mathbb{Z}/r\mathbb{Z}$ and $e(P, aQ) = z$. Hence, the solution to the original FAPI-2 problem is $aP = \psi_2(Q')$.

Showing $(3) \Rightarrow (1)$ is similar. $\square$

The above results imply a refinement of Verheul's result which shows that one does not necessarily have to solve both FAPI-1 and FAPI-2 to get interesting applications.

Corollary 7: Let $e : G_1 \times G_2 \to G_T$ be a non-degenerate bilinear pairing on cyclic groups of prime order r. Suppose there is an efficiently computable (i.e., can be computed in polynomial time) homomorphism $\psi_2 : G_2 \to G_1$. If one can solve FAPI-1 in polynomial time then one can solve the computational Diffie-Hellman problem in G_1, G_2 and G_T in polynomial time.

By symmetry we get:

Corollary 8: Let $e : G_1 \times G_2 \to G_T$ be a non-degenerate bilinear pairing on cyclic groups of prime order r. Suppose there is an efficiently computable homomorphism

$\psi_1 : G_1 \rightarrow G_2$. If one can solve FAPI-2 in polynomial time then one can solve the computational Diffie-Hellman problem in G_1, G_2 and G_T in polynomial time.

The existence of efficiently computable homomorphisms $\psi_i : G_i \rightarrow G_{3-i}$ (i.e., "distortion maps") depends on the curve and groups G_1 and G_2. For elliptic curves with $k > 1$, we have the following: except for elements of its two eigenspaces, the Frobenius endomorphism φ can always be used to construct a distortion map. Distortion maps for the 1-eigenspace exist if and only if the curve is supersingular [23], [10]. Similarly, one can show that for the q-eigenspace, distortion maps exist if and only if the curve is supersingular.

We now introduce some variants of the bilinear-Diffie-Hellman problem.

Definition 9: Let $e : G_1 \times G_2 \rightarrow G_T$ be a non-degenerate bilinear pairing.
The **bilinear-Diffie-Hellman problem (BDH-1)** is: given $P, aP, bP \in G_1$ and $Q \in G_2$ to compute $e(P, Q)^{ab}$.
The **bilinear-Diffie-Hellman problem (BDH-2)** is: given $P \in G_1$ and $Q, aQ, bQ \in G_2$ to compute $e(P, Q)^{ab}$.

We now show that it is sufficient to solve FAPI-1 to solve BDH-1. Hence we have BDH-1 $\leq$ FAPI-1 $\leq$ CDH in G_1 $\leq$ (FAPI-1 and FAPI-2). We stress that the pairing in the statement of FAPI-1 does not need to be the same pairing as in the statement of BDH-1.

Corollary 10: Suppose one can solve FAPI-1 in polynomial time, then one can solve BDH-1 in polynomial time.
PROOF: Suppose we are given a BDH-1 instance (P, aP, bP, Q). Using an oracle to solve FAPI-1 one can compute a group homomorphism $\psi : G_1 \rightarrow G_2$ such that $\psi(P) = Q$. Hence, one can compute $aQ = \psi(aP)$ and obtain $z = e(bP, aQ) = e(P, Q)^{ab}$. $\square$

Corollary 11: Suppose one can solve FAPI-2 in polynomial time, then one can solve BDH-2 in polynomial time.

It follows that if one can solve only one of the two pairing inversion problems, then there are potential weaknesses for pairing-based cryptosystems. Note that in this case, cryptography in subgroups of finite fields does not seem to be affected.

Let $P, aP \in G_i$ for $i = 1$ or 2. Mireles [17] has observed that if one can solve FAPI-i then one has a static Diffie-Hellman oracle $O(Q) = aQ$ for $Q \in G_i$. Hence one can apply results of Brown and Gallant [4] and Cheon [5] to speed up the computation of the discrete logarithm problem in G_i. More precisely, if $l < r^{1/2}$ is a divisor of $r - 1$ then with l calls to the FAPI-i oracle one can compute a in $O(\sqrt{r/l})$ group operations in G_i. Similarly, if $l < r^{1/3}$ is a divisor of $r + 1$ then with l calls to the FAPI-i oracle one can compute a in $O(\sqrt{r/l})$ group operations in G_i.

As a final application we observe that some cryptosystems can be broken directly using one-sided pairing inversion. For example, the identity-based signature scheme of Hess [13] has signature (u, v) on message m such that $v = H_2(m, e(u, P)e(H_1(ID), -Q_{TA})^v)$. If one can solve FAPI-2 then one can forge signatures by choosing a random element $z \in G_T$, setting $v = H_2(m, z)$ and then solving for u the equation $e(u, P) = ze(H_1(ID), Q_{TA})^v$.

IV. PAIRINGS

We recall some background on pairings. Let $\mathcal{C}$ be a non-singular projective curve of genus g over $\mathbb{F}_q$. We denote by $\text{Pic}^o_{\mathbb{F}_q}(C)$ the degree zero divisor class group of C over $\mathbb{F}_q$. Divisor classes are denoted by $\overline{D}$.

Let r be coprime to q. It is typical for cryptographic applications to take r a (large) prime divisor of $\#\text{Pic}^0_{\mathbb{F}_q}(\mathcal{C})$. It is often the case that $r \approx q^g$, but in some situations it is necessary to take r smaller. The embedding degree is defined to be the smallest positive integer k such that $r \mid (q^k - 1)$. Note that the embedding degree is a function of q and r. The subgroup of r-th roots of unity of $\mathbb{F}_{q^k}^{\times}$ is denoted by $\mu_r = \{z \in \mathbb{F}_{q^k}^{\times} : z^r = 1\}$.

A. Tate-Lichtenbaum pairing

The Tate-Lichtenbaum pairing [20], [15], [9], [6] is defined to be a non-degenerate bilinear map

$$\text{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C})[r] \times \text{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C})/r\text{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C}) \longrightarrow \mathbb{F}_{q^k}^{\times}/(\mathbb{F}_{q^k}^{\times})^r$$

which is denoted $\langle \overline{D}_1, \overline{D}_2 \rangle_r$.

For many cryptographic applications we assume that $\text{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C})$ contains no elements of order r^2 (so we may identify $\text{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C})[r]$ with $\text{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C})/r\text{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C})$) and we consider the reduced Tate-Lichtenbaum pairing

$$t(\overline{D}_1, \overline{D}_2) = \langle \overline{D}_1, \overline{D}_2 \rangle_r^{(q^k-1)/r}.$$

The mathematical definition of the Tate-Lichtenbaum pairing is as follows. The argument on the left hand side of the Tate-Lichtenbaum pairing is represented by an $\mathbb{F}_{q^k}$-rational divisor D_1 of degree zero. Since $\overline{D}_1$ is a divisor class of order r, there is a function f_{r,D_1} with divisor

$$\text{div}(f_{r,D_1}) = rD_1.$$

The argument of the right hand side of the Tate-Lichtenbaum pairing can be represented by an $\mathbb{F}_{q^k}$-rational divisor D_2 of degree zero such that the supports of D_1 and D_2 are disjoint. Then the Tate-Lichtenbaum pairing is defined to be

$$\langle \overline{D}_1, \overline{D}_2 + r\text{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C}) \rangle_r = f_{r,D_1}(D_2) = \prod_P f_{r,D_1}(P)^{v_P(D_2)}.$$

Finally, we note that $f_{r,D}$ with $\text{div}(f_{r,D}) = rD$ is only defined up to scalar multiples from $\overline{\mathbb{F}}_q^{\times}$. It is possible to find $f_{r,D}$ which is defined over the field of definition of D and we assume this in the following. We will need to impose some additional normalisation conditions on $f_{r,D}$ later.

B. Ate pairings

For cryptographic purposes one applies one further simplification to the reduced Tate-Lichtenbaum pairing by restricting the pairing to certain cyclic subgroups G_1 and

G_2 of $\mathrm{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C})[r]$ that are Frobenius eigenspaces. Write φ for the q-power Frobenius map on $\mathcal{C}$ and the Frobenius endomorphism on $\mathrm{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C})$. Then we define

$$G_1 = \mathrm{Pic}^0_{\mathbb{F}_q}(\mathcal{C})[r], \qquad (1)$$

for which the eigenvalue of φ is 1. We also define

$$G_2 = \mathrm{Pic}^0_{\mathbb{F}_{q^k}}(\mathcal{C})[r] \cap \ker(\varphi - q). \qquad (2)$$

B.1 Ate pairings on elliptic curves

Let $\mathcal{E}$ be an ordinary elliptic curve over $\mathbb{F}_q$. Let t be the trace of the q-power Frobenius endomorphism φ of $\mathcal{E}$, such that $\#\mathcal{E}(\mathbb{F}_q) = q - t + 1$. We assume that $r \geq 5$ is a sufficiently large prime factor of $\#\mathcal{E}(\mathbb{F}_q)$ and that k is minimal such that $r \mid (q^k - 1)$.

If $P \in \mathcal{E}(\overline{\mathbb{F}}_q)$ has order r, then $(P) - (\infty)$ is a divisor of degree zero representing a divisor class of order r. For $P \in \mathcal{E}(\overline{\mathbb{F}}_q)$ of arbitrary order and any integer s we denote by $f_{s,P}$ a rational function on $\mathcal{E}$, defined over the field of definition of P, satisfying $\mathrm{div}(f_{s,P}) = s((P) - (\infty)) - ((sP) - (\infty))$. We also need to normalise $f_{s,P}$ as follows. Let $u \in \mathbb{F}_q(\mathcal{E})$ be a local uniformizer at ∞, that is z satisfies $u_\infty(z) = 1$. Then we define $\mathrm{lc}_\infty(f_{s,P}) = (u^{-v_\infty(f_{s,P})} f_{s,P})(\infty)$ and $f_{s,P}^{\mathrm{norm}} = f_{s,P}/\mathrm{lc}_\infty(f_{s,P})$. The function $f_{s,P}^{\mathrm{norm}}$ is defined over the field of definition K of P and is uniquely determined by s and P up to non-zero sth-power multiples from K.

Theorem 12: ([14], [16]) Let S be an integer with $S \equiv q \bmod r$. Define $N = \gcd(S^k - 1, q^k - 1)$ and $L = (S^k - 1)/N$. Let $c_S = \sum_{i=0}^{k-1} S^{k-1-i} q^i \bmod N$. Then

$$a_S : G_2 \times G_1 \to \mu_r, \quad (Q, P) \mapsto f_{S,Q}^{\mathrm{norm}}(P)^{c_S(q^k-1)/N}$$

defines a bilinear pairing, called the *elliptic ate pairing*. If $k \mid \#\mathrm{Aut}(\mathcal{E})$ then

$$a_S^{\mathrm{twist}} : G_1 \times G_2 \to \mu_r, \quad (P, Q) \mapsto f_{S,P}(Q)^{c_S(q^k-1)/N}$$

also defines a bilinear pairing, called the *twisted ate pairing*. Both pairings a_S and a_S^{twist} are non-degenerate if and only if $r \nmid L$.

The relation with the reduced Tate-Lichtenbaum pairing is

$$a_S(Q, P) = t(Q, P)^L \quad \text{and} \quad a_S^{\mathrm{twist}}(P, Q) = t(P, Q)^L.$$

We remark that the condition $k \mid \#\mathrm{Aut}(\mathcal{E})$ holds true if and only if $\mathcal{E}$ admits a twist of degree k. We say that $\mathcal{E}$ admits a twist of degree d if there is an elliptic curve $\mathcal{E}'$ defined over $\mathbb{F}_q$ and an isomorphism $\psi : \mathcal{E}' \to \mathcal{E}$ defined over $\mathbb{F}_{q^d}$, and d is minimal with this property. If $k \mid \#\mathrm{Aut}(\mathcal{E})$ does not hold one may still apply the theorem for a divisor e of k using a base extension of degree k/e.

One can take $S = q$ in Theorem 12, but the usual choice is $S = t - 1$, which has half the bit length of $\#\mathcal{E}(\mathbb{F}_q)$ and thus yields half the loop length of the standard reduced Tate-Lichtenbaum pairing, if $r \approx q$. Note that if $S < 0$ we compute $f_{S,Q}$ as

$$f_{S,Q} = (f_{-S,Q} v_{SQ})^{-1},$$

where $v_{SQ} = x - x_{SQ}$ is the vertical line through SQ. In certain cases it may be possible to choose S strictly smaller than $t - 1$, which yields an even more efficient computation [16].

The Duursma-Lee pairing [7] and the η_T-pairing from [1] can be regarded as a special form of the twisted ate pairing on supersingular elliptic curves.

B.2 Ate pairings on hyperelliptic curves

For hyperelliptic curves the situation is somewhat different. In order to formulate the main results from [12], we fix some notation. Let $\mathcal{C}$ be a hyperelliptic curve with a single point ∞ at infinity. For any divisor class $\overline{D}$ we denote by $\rho(\overline{D})$ the unique reduced divisor in $\overline{D}$ and by $\epsilon(\overline{D})$ the effective part of $\rho(\overline{D})$ so that we have $\rho(\overline{D}) = \epsilon(\overline{D}) - \deg(\epsilon(\overline{D}))(\infty)$. We apply the same normalisation to the function $f_{s,D}$ as above, namely $f_{s,D}^{\mathrm{norm}} = f_{s,D}/(\mathrm{lc}_\infty(f_{s,D}))$ for $\mathrm{lc}_\infty(f_{s,D}) = (u^{-v_\infty(f_{s,D})} f_{s,D})(\infty)$ and $u \in \mathbb{F}_q(\mathcal{C})$ a local uniformizer at ∞ over $\mathbb{F}_q$. A curve is called superspecial if its Jacobian is isomorphic to E^g with E a supersingular elliptic curve. The Jacobian of superspecial curves is hence also supersingular, and in particular has p-rank zero.

Theorem 13: ([12]) With the above notation and assumptions,

$$a : G_2 \times G_1 \to \mu_r : (\overline{D}_2, \overline{D}_1) \mapsto f_{q,\rho(\overline{D}_2)}^{\mathrm{norm}}(\epsilon(\overline{D}_1))$$

defines a non-degenerate, bilinear pairing called the *hyperelliptic ate pairing*. If $\mathcal{C}$ is superspecial and $d = \gcd(k, q^k - 1)$ then

$$\hat{a} : G_1 \times G_2 \to \mu_r : (\overline{D}_1, \overline{D}_2) \mapsto f_{q,\rho(\overline{D}_1)}^{\mathrm{norm}}(\epsilon(\overline{D}_2))^d$$

defines a non-degenerate, bilinear pairing.

If in any of the two pairings we have $\mathrm{supp}(\epsilon(\overline{D}_i)) \cap \mathrm{supp}(\rho(\overline{D}_j)) \neq \emptyset$, then $\epsilon(\overline{D}_i)$ needs to be replaced by any $D \in \overline{D}_i$ with $\mathrm{supp}(D) \cap \mathrm{supp}(\rho(\overline{D}_j)) = \emptyset$.

The relation with the reduced Tate-Lichtenbaum pairing is

$$t(\overline{D}_2, \overline{D}_1) = a(\overline{D}_2, \overline{D}_1)^{kq^{k-1}}$$

and

$$t(\overline{D}_1, \overline{D}_1) = \hat{a}(\overline{D}_1, \overline{D}_2)^{(k/d)q^{k-1}}.$$

One feature of the hyperelliptic ate pairing is that the final exponentiation is very simple.

V. RESTATEMENT OF THE PROBLEMS

In Section II, the problems FAPI-1 and FAPI-2 were defined in the setting of pairings on cyclic groups. In practice, pairings are often defined on a larger object than a cyclic group, and it may be sufficient for some applications to solve pairing inversion problems with respect to this larger domain. Hence, we give some more general definitions which are more suitable when discussing the Tate-Lichtenbaum pairing.

We now assume that a pairing e is a well defined, bilinear map (not necessarily non-degenerate)

$$e : G_1 \times G_2 \to \mu_r \subseteq \mathbb{F}_{q^k}^* \qquad (3)$$

where r is a large prime, G_1, G_2 are subgroups of $\mathrm{Pic}_C^0(\mathbb{F}_{q^k})$ with $q^k - 1 \equiv 0 \bmod r$. In particular, G_1 and G_2 are no-longer necessarily cyclic or of exponent r (which is why we cannot assume non-degeneracy).

We further assume that e is computed on divisor classes $\overline{D}_1$ and $\overline{D}_2$, represented by suitable divisors D_1 and D_2 with $\mathrm{supp}(D_1) \cap \mathrm{supp}(D_2) = \emptyset$, as

$$e(\overline{D}_1, \overline{D}_2) := (f_{s,D_1}(\epsilon(D_2)))^d , \qquad (4)$$

where s and d are integers and f_{s,D_1} a function with divisor $sD_1 - [s]D_1$. Here $[s]D_1$ denotes the unique reduced divisor equivalent with sD_1. Note that s is typically r, q or $S = t - 1$ (see above) and d is the required final exponentiation.

Note that since r is prime and e bilinear, we have either $e(\overline{D}_1, G_2) = \{1\}$ or $e(\overline{D}_1, G_2) = \mu_r$. In the former case, e is called degenerate for $\overline{D}_1$, whereas in the latter case e is called non-degenerate for $\overline{D}_1$.

Fixed Argument Pairing Inversion 1 (FAPI-1): Given a pairing e, a divisor class $\overline{D}_1 \in G_1$ such that $e(\overline{D}_1, G_2) = \mu_r$ and $z \in \mu_r$, compute $\overline{D}_2 \in G_2$ such that $e(\overline{D}_1, \overline{D}_2) = z$.

Fixed Argument Pairing Inversion 2 (FAPI-2): Given a pairing e, $\overline{D}_2 \in G_2$ such that $e(G_1, \overline{D}_2) = \mu_r$ and $z \in \mu_r$, compute $\overline{D}_1 \in G_1$ such that $e(\overline{D}_1, \overline{D}_2) = z$.

The above problems generalise the case when G_1 and G_2 are cyclic groups. In some situations it is possible to apply homomorphisms from G_1 and G_2 to cyclic subgroups, in which case many of the results of Section III can be applied in the more general setting.

Let $i = 1$ or 2 and choose a divisor $\overline{D}_{3-i} \in G_{3-i}$. It follows that any efficient algorithm to solve FAPI-$(3 - i)$ immediately leads to a computable group homomorphism $h_i : \mu_r \to G_i/K_i$, with K_i the kernel of the pairing for fixed $\overline{D}_{3-i}$. Furthermore, note that the homomorphism h_i is always injective. However, it may not be possible in practice to efficiently compute generators for K_i or G_i/K_i.

Miller inversion (MI). Let D_1 be fixed and let S be a set of divisors. Let $z \in \mathbb{F}_{q^k}^*$. Compute a divisor $D_2 \in S$ such that $z = f_{s,D_1}(D_2)$ or if no such divisor exists then output 'no solution'.

It is natural to try to invert a pairing $e(D_1, D_2) = f_{s,D_1}(D_2)^d$ (for some divisors D_1, D_2 and integers s, d) by first inverting the final exponentiation (i.e., taking d-th roots in the finite field) and then inverting the pairing function (Miller inversion). We discuss inverting Miller's algorithm in the next section.

VI. Inverting Miller's algorithm

The aim of this section is to show that inverting Miller's algorithm is not necessarily difficult.

Consider the elliptic ate pairing on $G_2 \times G_1$. Then one has $a_S(Q, P) = f_{S,Q}(P)^d$ for certain $d \in \mathbb{Z}$ and $S = t - 1$ where t is the trace of Frobenius (as noted earlier, when $S < 0$ one computes $f_{S,Q}$ as $1/(f_{-S,Q}v_{SQ})$). Note that S can be very small (e.g., $S = \pm 2$). One can explicitly compute the rational function $f_{S,Q}(x, y)$ for $Q \in G_2$, $Q \neq \infty$.

Lemma 14: Suppose $S \geq 2$ and that Q has order > 2. Then the functions $f_{S,Q}(x, y)$ and $f_{-S,Q}(x, y)^{-1}$ can be written in the form

$$f_{S,Q}(x, y) = (f_1(x) + y f_2(x))/(x - x_{SQ}) \qquad (5)$$
$$f_{-S,Q}(x, y)^{-1} = f_1(x) + y f_2(x) \qquad (6)$$

where $\deg(f_1(x)) \leq (S + 1)/2$ and $\deg(f_2(x)) \leq S/2 - 1$. PROOF: First, note that $f_{S,Q}$ has poles only at SQ and ∞, and the pole at SQ has multiplicity one, so $f_{S,Q}$ can be written in the form (5).

For $P \in E$ write $v_P(f(x, y))$ for the valuation of the function $f(x, y)$ at P. Then, by definition, $v_Q(f_1(x) + y f_2(x)) = S$ and $v_{-SQ}(f_1(x) + y f_2(x)) = 1$, and for all affine points $P \neq Q, -SQ$ one has $v_P(f_{S,Q}(x, y)) = 0$.

Suppose E has equation $y^2 = x^3 + Ax + B$. Then

$$\begin{aligned} S + 1 &= v_Q(f_1 + y f_2) + v_{-SQ}(f_1 + y f_2) \\ &= v_Q(N(f_1 + y f_2)) + v_{-SQ}(N(f_1 + y f_2)) \\ &= \deg_x(f_1(x)^2 - (x^3 + Ax + B)f_2(x)^2) \\ &= \max\{2 \deg_x(f_1(x)), 3 + 2 \deg_x(f_2(x))\} \end{aligned}$$

where $N(\alpha)$ is the norm of α with respect to the quadratic extension $k(x, y)/k(x)$ of function fields. It follows that $\deg_x(f_1(x)) \leq (S + 1)/2$ and $\deg_x(f_2(x)) \leq S/2 - 1$.

Equation (6) finally follows from $f_{-S,Q}^{-1} = (x - x_{SQ})f_{S,Q}$. $\square$

Let z be a target element of the finite field. If $S > 0$ then one can clear denominators (i.e., obtain $f_1(x) + y f_2(x) - z(x - x_{SQ})$) and, taking a resultant with the elliptic curve equation $F(x, y) = y^2 - x^3 - ax - b = 0$, one obtains the polynomial $(f_1(x) - z(x - x_{SQ}))^2 - f_2(x)^2(x^3 + ax + b)$ in x of degree at most $S + 1$. In the case $S < 0$ we recommend first computing z^{-1} and then solving $f_{-S,Q}v_{SQ} = z^{-1}$ in a similar fashion.

Hence, the problem of Miller inversion is reduced to finding the unique root in $\mathbb{F}_q$ of a polynomial $P \in \mathbb{F}_{q^k}[x]$ of degree $O(|t|)$. Equivalently, this is computing $\gcd(x^q - x, P(x))$, which can be done in $O(|t|^2 \log q)$ operations in $\mathbb{F}_{q^k}$ or equivalently $O(|t|^2 k^2 (\log q)^3)$ bit-operations.

Hence, as long as $|t|$ and k grow as a polynomial function of $\log r$, one can solve MI in polynomial time. We now show the existence of parameters for which this can occur. We call parameters (r, q, k) pairing friendly (with respect to a given security parameter κ) if r is a prime, q is a prime power and there is an elliptic curve E over $\mathbb{F}_q$ with order divisible by r and embedding degree k, and such that the discrete logarithm problem in $E(\mathbb{F}_q)[r]$ and the discrete logarithm problem in $\mu_r \subset \mathbb{F}_{q^k}^*$ cannot be solved using known methods in time less than 2^κ.

Lemma 15: There exist families of parameters of pairing friendly curves for which the Miller inversion problem can be solved in polynomial time.

PROOF: To balance the security of the DLP in G_i with the DLP in the finite field $\mathbb{F}_{q^k}$, we need that $r^{1/2} \approx L_{q^k}(1/3; c)$ with $c < 2$ the constant appearing in the Number or Function Field Sieve complexities. Let $\rho = \log q / \log r$, then in practice one often restricts to $\rho \leq 2$, which implies that k has to grow. Balancing the security levels, implies that k has to grow as

$$k \approx \alpha(c, \rho) \left(\frac{\log r}{\log \log r} \right)^2, \tag{7}$$

with α a constant depending on c and ρ. In practice, the approximation $\alpha \approx 1/(100\rho)$ seems adequate. From $\Phi_k(t - 1) \equiv 0 \bmod r$ follows that t can be as small as $r^{1/\varphi(k)}$. Furthermore, $\sqrt{k} \leq \varphi(k) \leq k$ for all k except $k = 2, 6$ and for large k we have

$$\liminf_{k \to \infty} \varphi(k) \frac{\log \log k}{k} = e^{-\gamma},$$

with $\gamma \approx 0.57721$ Euler's constant (see [18, pp. 319-320] for this formula). Using the bound $\sqrt{k} \leq \varphi(k)$, we conclude that the smallest trace of Frobenius grows as

$$t \approx \exp\left(\frac{\log r}{\varphi(k)} \right) \leq \exp\left(\frac{\log \log r}{\sqrt{\alpha(c, \rho)}} \right) = (\log r)^{1/\sqrt{\alpha(c, \rho)}}, \tag{8}$$

which is polynomial in $\log r$. $\square$

In fact, the better approximation $e^{-\gamma} k / \log \log k$ shows that t is allowed to grow much faster than its minimal value, basically as $O(t_{\min}^{\log r})$, so MI is a polynomial time problem for a much larger class of curves than only those with minimal t.

To construct curves with small t one can use a family of curves, such as those proposed by Brezing and Weng [3] with $t = x + 1$, $r = \Phi_k(x)$ for some $x \in \mathbb{Z}$. Alternatively one can use an adaptation of the Cocks-Pinch algorithm as follows. Given k, a discriminant D and a bound B_t on t, repeat the following: for all $t \in [-B_t, B_t]$, set $x = t - 1$ and choose a (large) prime number r with $r \mid \Phi_k(x)$ and $k \mid (r - 1)$. If D is a square mod r, compute $f_0 \equiv \pm(t - 2)/\sqrt{D} \bmod r$ and test if $s_j = t^2 - D(f_0 + jr)^2$ for $j = 0, \pm 1, \pm 2, \ldots$ can be written as $4p$ with p prime. If this is the case, output r and p. Of course, in practice, we only test if $\Phi_k(x)$ has a large prime factor r using trial division and a primality test.

Example 16: We give an example with a simplest possible pairing function. The elliptic curve $E : y^2 = x^3 + 4$ over $\mathbb{F}_q$ where q is the prime 41761713112311845269 has $t = -1$, $r = 715827883$, $k = 31$ and $D = -3$. Now $S = -2$ and as a Miller function we can take $1/f_{S,Q}^{\text{norm}}$ instead of $f_{S,Q}^{\text{norm}}$. As in Lemma 14, we see that $1/f_{S,Q}^{\text{norm}}$ is just the tangent to E at Q and hence is of the form

$$1/f_{S,Q}^{\text{norm}} = y - \lambda x - \nu$$

with $\lambda = 3x_Q^2/(2y_Q)$ and $\nu = (-x_Q^3 + 8)/(2y_Q)$.

One easily checks that the non-degeneracy conditions of Theorem 12 are satisfied and that the final exponentiation (after taking a greatest common divisor with $q^k - 1$) is equal to $(q^k - 1)/(3r)$. So

$$(Q, P) \mapsto \left(y_P - \lambda x_P - \nu \right)^{(q^k - 1)/(3r)}$$

indeed defines a non-degenerate pairing.

The problem of Miller inversion for some $z \in \mathbb{F}_{q^k}$ then boils down to computing the greatest common divisor of $x^q - x$ with

$$x^3 + 4 - (\lambda x + \nu + z)^2,$$

giving the x-coordinate of P. The y-coordinate of P is obtained by taking the square root of $x_P^3 + 4$ and checking the result. To give some idea of the running times for this using Magma, the gcd computation takes only a fraction of a second while the square root computation takes about 1-2 seconds.

For larger traces t the Miller inversion via the greatest common divisor computation of the resultant and $x^q - x$ quickly becomes ineffective due to the large degrees. It will then be faster to apply the final exponentiation and use the Pollard methods in $\mu_r \subseteq \mathbb{F}_{q^k}^\times$ or index calculus in $\mathbb{F}_{q^k}^\times$ to invert the pairing via a discrete logarithm computation.

VII. PAIRING INVERSION

In this section we consider ways to invert pairings. One approach is to take a suitable d-th root and then do Miller inversion; this is a 'two step' method. An alternative way to proceed is to try to invert the pairing in a single step. There seems to be a significant difference between FAPI-1 and FAPI-2. For example, to solve FAPI-1 for a fixed divisor D_1 one can express $f_{s,D_1}(D_2)^d$ as a rational function with indeterminates corresponding to the divisor D_2 (e.g., with the coefficients in the Mumford representation of D_2 being variables). For elliptic curves, the degree of $f_{s,D_1}(D_2)^d$ grows as sd. On the other hand, for FAPI-2 one can express $f_{s,D_1}(D_2)^d$ for fixed D_2 as a rational function; for elliptic curves the degree grows as $s^2 d$.

We first consider a special case which is of interest. Then we analyze the precise relation between FAPI-1 and MI, i.e. when does one problem polytime reduce to the other. Later we consider the problem of inverting a pairing in one step (rather than first inverting the final exponentiation and then inverting Miller's algorithm).

A. FAPI-1 for the ate pairing on special curves

As seen in Section VI there are cases of the ate pairing where it is easy to invert Miller's algorithm and it is natural to try to invert pairings in this case.

We revisit Example 1 in this context. As discussed, a non-degenerate bilinear pairing can be computed as

$$a_2(Q, P) = \left(y_P - \lambda x_P - \nu \right)^{(q^k - 1)/(3r)}.$$

To solve FAPI-1 we are given $Q = (x_Q, y_Q)$ and a target $z \in \mu_r \subseteq \mathbb{F}_{q^k}^*$. One can compute $\lambda = (3x_Q^2)/(2y_Q)$ and

$\nu = (-x_Q^3 + 8)/(2y_Q)$. It suffices to find $x, y \in \mathbb{F}_q$ such that

$$(y - \lambda x - \nu)^{(q^k-1)/(3r)} = z.$$

The main problem is that there are $d = (q^k - 1)/(3r)$ possible roots of z and only one of them is likely to be of the correct form $y - \lambda x - \nu$ for some $(x, y) \in E(\mathbb{F}_q)$. It is easy to compute random d-th roots of z, but it seems to be hard to select the correct root efficiently. For further discussion see [21].

Note that one can obtain further equations with the same solution (x, y) from $a_2(uQ, P) = z^u$ for any $1 \le u < r$.

The problem FAPI-1 is seen to be similar to the following more elementary problem: Suppose we are given many pairs $(a, z) \in \mathbb{F}_{q^k}^2$, such that $z = (a + x)^d$ for some unknown value $x \in \mathbb{F}_q$, to find x. Usually a small number of pairs suffice to determine x uniquely, but it appears a hard problem to actually compute it when d is a large divisor of $(q^k - 1)$.

B. Is FAPI-1 $\le_P$ MI?

The conventional wisdom is that FAPI-1 is strictly harder than MI, since the final exponentiation destroys information. Precisely, given a pairing value $z \in \mu_r$, one knows that $f_{s,D_1}(D_2)^d = z$, but there are d possibilities for the value $f_{s,D_1}(D_2)$. One might think that the attacker has to try inverting f_{s,D_1} for all d roots in turn, which would be infeasible if d is large. In this section we show this reasoning to be fundamentally flawed for the Tate-Lichtenbaum pairing. We will show that, in most cases, it suffices to choose a random d-th root of z. The situation for the ate pairing is subtly different and we discuss this case at the end of the next subsection.

Definition 17: Let $e : G_1 \times G_2 \to \mu_r$ be a pairing as above and let D_1 be a divisor representing an element of G_1. Define $S_1(D_1, G_2)$ to be the set of all divisors D_2 corresponding to elements of G_2 for which the pairing $e(\overline{D}_1, \overline{D}_2)$ can be computed as $f_{s,D_1}(D_2)^d$.

For example, for the Tate-Lichtenbaum pairing $t : E(\mathbb{F}_q)[r] \times E(\mathbb{F}_{q^k}) \to \mathbb{F}_{q^k}^*$ and for $D_1 = P \in G_1 = E(\mathbb{F}_q)[r]$ we have $G_2 = E(\mathbb{F}_{q^k})$ and

$$E(\mathbb{F}_{q^k}) - \langle P \rangle \subseteq S_1(D_1, G_2) \subseteq E(\mathbb{F}_{q^k}).$$

On the other hand, for the ate or twisted ate pairing we have G_1 and G_2 as in equations (1) and (2) so G_2 is much smaller than in the Tate-Lichtenbaum case. We have $S_1(D_1, G_2)$ equals

$$\left\{ \epsilon(\overline{D}_2) \mid \overline{D}_2 \in G_2 \text{ and } \operatorname{supp}(D_1) \cap \operatorname{supp}(\epsilon(\overline{D}_2)) = \emptyset \right\},$$

where D_1 is of the form $\rho(\overline{D}_1)$ for $\overline{D}_1 \in G_1$.

Note that if $D_2, D_3, D_4 \in S_1(D_1, G_2)$ are such that D_4 is equivalent to $D_2 + D_3$ then it follows that

$$f_{s,D_1}(D_2)^d f_{s,D_1}(D_3)^d = f_{s,D_1}(D_4)^d. \qquad (9)$$

In this section we consider when being able to invert Miller functions is sufficient for inverting pairings. It is

clear that if $d = 1$, or d is polynomially small, then FAPI-1 $\le_P$ MI: we invoke MI on each d-th root z' in turn until a solution is returned.

For large d, the situation is more interesting. For each $z_0 = e(\overline{D}_1, \overline{D}_2) \in \mu_r$ there are precisely d possible d-th roots z to choose from. It is no longer feasible to run MI on each root in turn. We know of no efficient algorithm to identify the roots that lie in $f_{s,D_1}(S_1(D_1, G_2))$. Hence, we propose choosing a random d-th root and then running MI. The following discussion estimates the success probability of this method.

Fix a divisor D_2 such that $z_0 = f_{s,D_1}(D_2)^d$. Note that if $D \in S_1(D_1, G_2)$ is such that $f_{s,D_1}(D) = z$ then $e(\overline{D}_1, \overline{D}) = e(\overline{D}_1, \overline{D}_2)$. Denote by $S_2(D_1)$ the set of all divisors D_3 such that $e(\overline{D}_1, \overline{D}_3) = 1$. Bilinearity implies that

$$\overline{D} \equiv \overline{D}_2 + \overline{D}_3 \text{ where } D_3 \in S_2(D_1). \qquad (10)$$

The number of choices for D is therefore equal to the number of divisors in $(\overline{D}_2 + S_2(D_1)) \cap S_1(D_1, G_2)$. If this intersection is sufficiently large and if the corresponding values for z are relatively evenly distributed, then there is a good chance that a divisor D exists such that $f_{s,D_1}(D) = z$ for a randomly chosen d-th root z of z_0. More research is needed to clarify this issue. In particular, it is necessary to understand the distribution of values z over all choices for D.

Conversely, if $|(\overline{D}_2 + S_2(D_1)) \cap S_1(D_1, G_2)|/d < 1/2^\kappa$ for sufficiently large κ then the probability of being able to solve Tate-Lichtenbaum pairing inversion by taking a random d-th root must be negligible.

Example 18: To illustrate this approach, we apply the above to the reduced Tate-Lichtenbaum pairing on elliptic curves. In this case, we have $G_1 = E(\mathbb{F}_q)[r]$, $G_2 = E(\mathbb{F}_{q^k})$ and $d = (q^k - 1)/r$. Let $D_1 \in G_1$ and $z_0 \in \mu_r \subseteq \mathbb{F}_{q^k}^*$. We want to determine the probability, for a randomly chosen d-th root z of z_0, that there is a divisor $D_2 \in G_2$ such that $f_{s,D_1}(D_2) = z$.

From the definition of the Tate-Lichtenbaum pairing follows that $rE(\mathbb{F}_{q^k}) \subset S_2(D_1)$. The size of $rE(\mathbb{F}_{q^k})$ is $\approx q^k/r^2$ and thus is much smaller than d, so this is not sufficient to argue that a random d-th root of z_0 is in the image of the pairing. However, for $k > 1$, Lemma IX.8 in [2] shows that $S_2(D_1)$ also contains $E(\mathbb{F}_{q^e})$ for all $e \mid k$. Since $r \| E(\mathbb{F}_q)$, we conclude that $E(\mathbb{F}_q)[r] \cap rE(\mathbb{F}_{q^k}) = \{O\}$ and thus

$$|S_2(D_1)| \ge |E(\mathbb{F}_q)[r]||rE(\mathbb{F}_{q^k})| \approx rq^k/r^2 \approx d.$$

This suggests that for the Tate-Lichtenbaum pairing with $k > 1$, we indeed have FAPI-1 $\le_P$ MI.

We remark that the above example does not imply that one can efficiently solve FAPI-1 for the Tate-Lichtenbaum pairing. We have shown that the final exponentiation is not an obstacle, but inverting Miller's algorithm is still hard since the degree of f_{s,D_1} is exponentially large.

For the ate pairing there is the further complication that $S_1(D_1, G_2)$ is typically very small. This is because the ate pairing is only bilinear on $G_1 \times G_2$ where G_1 and G_2 are

Frobenius eigenspaces. In practice one might be able to invert the Miller function to get a divisor corresponding to a divisor class outside G_2, but it is unclear that this has any usefulness since the pairing is not expected to be bilinear outside $G_1 \times G_2$.

C. Is MI $\leq_P$ FAPI-1?

Although MI looks easier than FAPI-1, the former does not necessarily polytime reduce to the latter. More precisely: does MI for f_{s,D_1} and $S_1(D_1, G_2)$ polytime reduce to FAPI-1 for e and $\overline{D}_1$, where we make the implicit assumption that $e(\overline{D}_1, \cdot)$ is computed as $f_{s,D_1}(\cdot)^d$. A possible reduction would be as follows: given a value $z \in f_{s,D_1}(S_1(D_1, G_2))$, call FAPI-1 on z^d, which returns a class $\overline{D}_2$ with $e(\overline{D}_1, \overline{D}_2) = z^d$.

So for $d = 1$, any element of $V := S_1(D_1, G_2) \cap \overline{D}_2$, where $\overline{D}_2$ is considered as a set, will be a solution to MI. In this case we do have MI $\leq_P$ FAPI-1. It is interesting to note that the cardinality of the set V can be very large, which directly follows from the fact that the pairing is well-defined. For instance: in case of the ate pairing on $G_2 \times G_1$ which has $d = 1$, the set V consists of all $\mathbb{F}_q$-rational degree zero divisors $D_2 \in \overline{D}_2$ with $\mathrm{supp}(D_1) \cap \mathrm{supp}(D_2) = \emptyset$.

For $d > 1$, there are two cases: the first case is where V only contains one element (or a polynomial number of elements), then again we have MI $\leq_P$ FAPI-1. Examples are: the ate pairing on $G_1 \times G_2$ or the elliptic ate pairing. In the second and most general case, the cardinality of V does not grow polynomially, so we cannot conclude that MI $\leq_P$ FAPI-1. The reduced Tate-Lichtenbaum pairing provides an example of this behaviour.

D. Degree bounds

We have seen that Miller functions can be of (very) low degree and hence easy to invert. In this section we look at the question whether pairing functions can have low degree as well.

The following lemma shows that it is not possible to find pairings on elliptic curves which can be inverted for the reason that the corresponding pairing functions have low degree. It thus provides a security argument for pairing based cryptography, to some extent at least.

Lemma 19: Let E be an elliptic curve and $f \in \mathbb{F}_{q^k}(E)$. Assume that $Q \mapsto f(Q)^d$ defines a non-constant homomorphism $G_2 \to \mu_r$ for some positive exponent d. Then $d \deg(f) \geq (1/6)\#G_2$.
PROOF: Let $h = (f^d \circ [2]) - f^{2d}$. Then $h(Q) = 0$ for all $Q \in G_2$. Since f is not constant, by considering the degrees of $f^d \circ [2]$ and f^{2d} it follows that h is not zero. Thus $\#G_2 \leq \deg(h) \leq 4\deg(f^d) + 2\deg(f^d) \leq 6d \deg(f)$.
□

More precisely, the lemma shows that even just homomorphisms cannot be obtained using small degree functions. We saw in subsection VII-B above that if f has small degree, then MI is easy. But Lemma 19 implies $d = \Omega(r)$ in this case, so d must be large and there will be many possible d-th roots of z to choose from. As we have seen, this appears to be the obstacle to pairing inversion.

If on the other hand d is small then it would be easy to iterate through all d-th roots. Lemma 19 then implies that f has large degree and so MI seems to be hard.

It is even hard to construct elliptic curves such that pairing inversion would be easy. One possible way of attack could be to arrange for f and d such that there is a map $h : \mu_r \to G_2$ given by polynomials or rational functions of "small degree" together with a final exponentiation. More generally, h should have a compact representation. Since there does not seem to be a Riemann-Roch theory available like for the pairing case $G_2 \to \mu_r$, it is unclear whether or how such a representation could be achieved. Note that polynomials for h can always be obtained via interpolation (see [19]), but this does not give a compact representation.

VIII. INVERTING PAIRINGS ON HIGH GENUS CURVES

A. General considerations

As we have seen, there does not seem to be much hope to efficiently invert pairings on elliptic curves. So the question arises whether it is possible to actually construct a curve of higher genus where pairing inversion (or more generally inversion of a homomorphism given by a rational function) is actually possible in a non-trivial situation.

There is some reason to expect success in this case, since the hyperelliptic ate pairing gives both significant loop shortening as well as a very simple final exponentiation. More precisely, this pairing takes the form

$$(D_1, D_2) \mapsto f_{q,\rho(D_1)}^{\mathrm{norm}}(\epsilon(D_2)).$$

So there is no final exponentiation and the degree of the function is $O(gq)$, which is polynomial in $\log(r)$ if q is fixed and g tends to infinity.

However, these potential simplifications are opposed by the fact that we are evaluating functions at divisors rather than a single point. The divisor $\epsilon(D_2)$ is in general a sum of g independent points. Thus FAPI-1 now becomes a problem of solving multivariate systems of equations.

By replacing D_2 with random multiples we can achieve that $\epsilon(D_2)$ consists of a Galois orbit of $\mathbb{F}_{q^{kg}}$-rational points, or put differently, represents an $\mathbb{F}_{q^k}$-rational place of degree g. That is, we only need to solve a multivariate system where the variables are conjugated under the q^k-power Frobenius. Since q^k is large, this leads to a univariate system of equations but with large degrees. Hence the situation eventually becomes similar to the elliptic curve case, from a complexity point of view.

We give some details of this approach in the next section for a very special family of curves. Our results do not currently imply any weakness for pairing inversion on these curves.

B. Duursma curves

Duursma and Lee [7] proposed implementing pairings on the supersingular curves

$$C : y^2 = x^p - x + b \qquad \text{where} \qquad b = \pm 1$$

over $\mathbb{F}_p$ where $p \equiv 3 \pmod 4$. These curves have genus $g = (p-1)/2$ and have embedding degree $k = 2p$. For a point $P = (x_P, y_P) \in C(\mathbb{F}_{p^m})$ one can show that the point $[p]P = (x_P^{p^2} + 2b, -y_P^{p^2})$ satisfies $p((P) - (\infty)) \equiv ([p]P) - (\infty)$ in the divisor class group.

Duursma and Lee showed how to compute pairings on these curves extremely efficiently. The key idea is to use, for $P = (x_P, y_P) \in C$, the function

$$g_P(x,y) = y_P^p y - (x_P^p - x + b)^{(p+1)/2}$$

which has divisor $(g_P) = p(P) + (-[p]P) - (p+1)(\infty)$. One can then compute eta or ate pairings on $C(\mathbb{F}_{p^m})$ efficiently. Galbraith, Ó hÉigeartaigh and Sheedy [11] showed that if one includes denominators then the final exponentiation is just a squaring. One can obtain suitable parameters with rather small values $\mathbb{F}_{p^m}$. For example, one gets parameters which could be secure by working over $\mathbb{F}_{47^2}$ or over $\mathbb{F}_{83}$.

Let us consider the $p = 83$ case (we do not claim that there is a large prime divisor of the group order in this case). One has $\#\mathrm{Pic}_C^0(\mathbb{F}_{83}) \approx 2^{262}$ and $k = 2p = 166$ so $\#\mathbb{F}_{83^k} \approx 2^{1058}$. Let $P, Q \in C(\mathbb{F}_{83})$. Then the pairing of P with $\psi(Q)$ (where ψ is the usual distortion map, see [7], [11]) can be computed as

$$z = (g_P(\psi(Q))/(x_{\psi(Q)} - x_P^{p^2} - 2b))^2.$$

To solve FAPI-1 for this pairing one tries each of the two square roots $z^{1/2}$ in turn, computes $f(x,y) = g_P(x,y) - z^{1/2}(x - x_P^{p^2} - 2b)$ and takes a resultant with $y^2 = x^p - x + b$ to get a polynomial in x of degree $p+1$. It is then a simple matter to find roots in $\mathbb{F}_p$.

However, it is not sufficient to be able to invert pairings on single points. As we have mentioned, we are usually pairing general divisors and for applications such as those in Section III it is necessary to be able to invert pairings in the general case. We now discuss the two intermediate cases $e(D_1, Q)$ and $e(P, D_2)$ in turn.

If $D_1 = \sum_{i=1}^g (P_i) - g(\infty)$, where the points P_i are typically defined over some extension of $\mathbb{F}_p$, then $e(D_1, Q) = \prod_{i=1}^g e(P_i, Q)$. The above method can still be used by taking the product of the g functions before taking the resultant. This leads to a univariate polynomial of degree $< p^2$ which is still feasible to solve. This may look like progress towards breaking some cryptographic protocols which are implemented using special divisors (see [1], [8] for such proposals) but in such cases we could also have broken the system simply by trying all $Q \in C(\mathbb{F}_p)$ (note that $\#C(\mathbb{F}_{83})$ is either 1 or 167).

If $D_2 = \sum_{i=1}^g (Q_i) - g(\infty)$, again with Q_i defined over some extension of $\mathbb{F}_p$, then one has $e(P, D_2) = \prod_{i=1}^g e(P, Q_i)$. If one introduces indeterminates $Q_i = (x_i, y_i)$ for each point then one obtains a large and under-determined multivariate system. The attack seems hopeless in this case.

As mentioned earlier, we can exploit the fact that with probability $1/g$ the divisor D_2 is of the form $D_2 = \sum_{i=1}^g (Q^{\sigma_i}) - g(\infty)$ where $Q \in C(\mathbb{F}_{p^g})$ and where

$\mathrm{Gal}(\mathbb{F}_{p^g}/\mathbb{F}_p) = \{\sigma_1, \ldots, \sigma_g\}$. One then notes that

$$e(P, \psi(D_2)) = \prod_{i=1}^g e(P, \psi(Q^{\sigma_i})).$$

Now, ψ is defined over $\mathbb{F}_{p^k}$ and $\gcd(k, g) = 1$ so we can write the above as

$$\prod_{i=1}^g e(P, \psi(Q)^{q^{ki}}) = \prod_{i=0}^{g-1} e(P, \psi(Q))^{p^{ki}}$$
$$= e(P, \psi(Q))^{1+p^k+\cdots+p^{k(g-1)}}$$

which is computing $N_{\mathbb{F}_{p^{kg}}/\mathbb{F}_{p^k}}(e(P, \psi(Q)))$. To solve FAPI-1 now simply requires inverting the norm and then following the previous method. The problem is now very similar the problem with the elliptic ate pairing; there are too many possible pre-images to try. Hence this approach also fails.

Solving FAPI-1 for the fully general case of $e(D_1, D_2)$ is also hard for the above reasons. It would be very interesting to have some new techniques to handle pairing inversion in this case.

IX. Conclusion

We have defined and analysed the fixed argument pairing inversion problems FAPI-1 and FAPI-2, as well as the Miller inversion problem (MI).

We have shown that MI can be easy, for example for the ate pairing with very small $|S|$. Such examples tend to have rather large k and inverting the final exponentiation seems to be hard. Hence the pairing inversion problems seem to be hard for these cases.

We have shown that for the Tate-Lichtenbaum pairing it is sufficient to take a random root when inverting the final exponentiation. Hence we have shown FAPI-1 $\leq$ MI for this pairing (where FAPI-1 is the more general version of Section V so that the group G_2 is not cyclic). Note that MI seems to be hard for the Tate-Lichtenbaum pairing.

We have also shown examples (such as the hyperelliptic ate pairing) where inverting the final exponentiation is easy. We deduce that MI is equivalent to FAPI-1 for these examples. We also deduce MI $\leq$ FAPI-1 for the ate pairing and elliptic ate pairing on cyclic groups. However, pairing inversion seems to be hard for all these examples.

To conclude, our methods do not solve the pairing inversion problem for any cryptographically useful curves. Hence, our results currently support the security of pairing-based cryptosystems.

Acknowledgements

References

[1] P. S. L. M. Barreto, S. Galbraith, C. O hEigeartaigh, and M. Scott. Efficient pairing computation on supersingular abelian varieties. *Designs, Codes and Cryptography*, Vol. 42, No. 3 (2007) 239–271.

[2] I. F. Blake, G. Seroussi, and N. P. Smart. *Advances in elliptic curve cryptography*. Cambridge, 2005.

[3] F. Brezing and A. Weng. Elliptic curves suitable for pairing based cryptography. *Designs, Codes and Cryptography*, 37, 133–141, 2005.

[4] D. R. L. Brown and R. P. Gallant, The static Diffie-Hellman problem, eprint archive 2004/306.

[5] J.-H. Cheon, Security analysis of the strong Diffie-Hellman problem, in S. Vaudenay (ed.), Eurocrypt 2006, Springer LNCS 4004 (2006) 1–11.

[6] H. Cohen, G. Frey, R. Avanzi, C. Doche, T. Lange, K. Nguyen, and F. Vercauteren. *Handbook of elliptic and hyperelliptic curve cryptography*. Discrete Mathematics and its Applications (Boca Raton). Chapman & Hall/CRC, Boca Raton, FL, 2006.

[7] I. M. Duursma and Hyang-Sook Lee. Tate Pairing Implementation for Hyperelliptic Curves $y^2 = x^p - x + d$. In Chi-Sung Laih, editor, *ASIACRYPT*, volume 2894 of *Lecture Notes in Computer Science*, pages 111–123. Springer, 2003.

[8] G. Frey and T. Lange. Fast Bilinear Maps from the Tate-Lichtenbaum Pairing on Hyperelliptic Curves. In F. Hess, S. Pauli, M. Pohst, editors, *ANTS VII*, volume 4076 of *Lecture Notes in Computer Science*, pages 466–479. Springer, 2006.

[9] G. Frey and H-G. Rück. A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. *Math. Comp.*, 62(206):865–874, 1994.

[10] S. D. Galbraith and V. Rotger, Easy decision Diffie-Hellman groups, LMS J. Comput. Math. 7 (2004) 201–218.

[11] S. Galbraith, C. O hEigeartaigh, and C. Sheedy, Simplified pairing computation and security implications, *J. Math. Crypt.*, **1** (2007) 267–281.

[12] R. Granger, F. Hess, R. Oyono, N. Thériault and F. Vercauteren. Ate pairing on hyperelliptic curves. In *Advances in Cryptology – EUROCRYPT 2007*, volume 4515 of *Lecture Notes in Computer Science*, pages 419–436. Springer-Verlag, 2007.

[13] F. Hess, Efficient identity based signature schemes based on pairings, In K. Nyberg and H. Heys, (eds.), SAC 2002, Springer LNCS 2595 (2000) 310–324.

[14] F. Hess, N. Smart, and F. Vercauteren. The Eta-pairing revisited. *IEEE Transactions on Information Theory*, 52(10):4595–4602, 2006.

[15] S. Lichtenbaum. Duality theorems for curves over p-adic fields. *Invent. Math.*, 7:120–136, 1969.

[16] S. Matsuda, N. Kanayama, F. Hess and E. Okamoto, Optimised versions of the Ate and Twisted Ate Pairings, in S. D. Galbraith (ed.), Cryptography and Coding, Springer LNCS 4887 (2007) 302–312.

[17] D. J. Mireles Morales, Cheon's algorithm, pairing inversion and the discrete logarithm problem, preprint.

[18] P. Ribenboim, The New Book of Prime Number Records, Springer, 1996.

[19] T. Satoh, On polynomial interpolations related to Verheul homomorphisms, *LMS. J. Comput. Math.*, **9** (2006) 135–158.

[20] J. Tate. WC group over 𝔭-adic fields. *Séminaire Bourbaki*, 1958.

[21] F. Vercauteren, The hidden root problem, in S. D. Galbraith and K. G. Paterson (eds.), Pairing 2008, Springer LNCS 5209 (2008) 89–99.

[22] E. Verheul. Evidence that XTR Is More Secure than Supersingular Elliptic Curve Cryptosystems. In B. Pfitzmann, editor, *EUROCRYPT*, volume of 2045 *Lecture Notes in Computer Science*, pages 195–210. Springer, 2001.

[23] E. Verheul, Evidence that XTR is more secure than supersingular elliptic curve cryptosystems, *J. Crypt.*, **17**, No. 4 (2004) 277–296.