Miller Inversion is Easy for the Reduced Tate Pairing on Trace Zero Supersingular Curves

Takakazu Satoh
e-mail: satoh.df603@gmail.com

Abstract

Let q be a power of an odd prime. (In practice, q itself is a large prime.) Denote the finite field of q elements by $\mathbf{F}_q$. We present a simple algorithm for Miller inversion for the reduced Tate pairing on supersingular elliptic curve defined over $\mathbf{F}_q$ of trace zero (i.e. ${}^\#E(\mathbf{F}_q)=q+1$). Assume we precomputed a generator of the 2-Sylow subgroup of $\mathbf{F}_q^\times$, which depends only on q. Then our algorithm runs deterministically with $O((\log q)^3)$ bit operations.

1. Introduction

Difficulty of pairing inversion is a fundamental assumption in pairing based cryptography. Duc and Jetchev[2, Sect. 5.2] gives explicit description how pairing inversions break Boneh-Franklin's IBE, Hess' IBS and Joux's tripartite key agreement protocol. More interestingly, Verheuel[9] proved that the computational Diffie-Hellman problem is reduced to pairing inversion. The result is extended to asymmetric pairings by Karabina, Knapp and Menezes[5].

Galbraith, Hess and Vercauteren[3] proposed a two step pairing inversion framework. The first step is called final exponentiation inversion (FEI), while the second step is called Miller inversion (MI). In general, both steps are considered to be difficult. However, [3, Sect. 6] proposes a family of pairing friendly elliptic curves whose MI are easy. Assuming Bateman and Horn conjecture[1] which is plausible but unproved, we see that the family consists of infinitely many elliptic curves. The purpose of this short note is to prove that MI is easy for trace zero supersingular curves. Our algorithm is simple but it seems that it has not observed before.

Let us be more specific. Let q be a power of an odd prime and put $r:=q^2$. We denote the q-th power Frobenius operator by φ_q. Let E be a supersingular elliptic curve over $\mathbf{F}_q$ defined by the Weierstrass model (this is important for our method). We assume E is of trace zero, i.e., ${}^\#E(\mathbf{F}_q)=q+1$. Let l be an odd number dividing $q+1$. Put $G_1:=E[l]\cap E(\mathbf{F}_q)$ and $G_0:=\{P\in E[l]:\varphi_q(P)=-P\}$. By Schoof[7, Lemma 4.8], $E(\mathbf{F}_r)=E[q+1]$ and $E(\mathbf{F}_q)$ is isomorphic to either $\mathbf{Z}/(q+1)\mathbf{Z}$ or $\mathbf{Z}/\left(\dfrac{q+1}{2}\right)\mathbf{Z}\oplus\mathbf{Z}/2\mathbf{Z}$. In particular, $\varphi_q^2|_{E[l]}=\mathrm{id}_{E[l]}$, which yields $G_0\cap G_1=\{\mathcal{O}\}$ and $E[l]=G_0\oplus G_1$ as an Abelian group. Moreover, G_1 is cyclic since l is an odd divisor of $q+1$. Then G_0 is also cyclic. Let $h_{q+1,A}$ be the $(q+1)$-st normalized Miller function. We consider the reduced Tate pairing defined by

$$e_{q+1}(A,Q) := h_{q+1,A}(Q)^{q-1}$$

which is a non-degenerate bilinear pairing on $G_0\times G_1$. The fixed argument pairing inversion problem (FAPI) is to find $Q\in G_1$ satisfying $e(A,Q)=z$ for given $z\in\mu_l-\{1\}$ and $A\in G_0-\{\mathcal{O}\}$, In the above two step framework, FEI returns the value of $h_{q+1,A}(Q)$ and

MI finds Q from the output of FEI. Under this circumstance, we give a simple algorithm for MI.

If we exclude side-channel attacks (and use of quantum computers), FEI seems to be a very hard problem. See Vercauteren[8]. If FEI is actually a hard problem, our result has probably no impact to real world cryptography. However, Lashermes, Fournier and Goubin[6] gives a fault attack method for FEI. Although their method is intended for ordinary curves, it is also applicable to supersingular curves. Indeed, the method described in Section 4.2 of [6] is sufficient for the embedding degree two case. Thus, if one has concerns about fault attacks, final exponentiation must be so implemented that it is immune to such attacks.

2. The algorithm

We keep notation in the previous section. Let ξ be the X-coordinate function. Our algorithm is as follows.

Algorithm 2.1.
Input: $v \in \mathbf{F}_r$, $A \in G_0 - \{\mathcal{O}\}$. // Note that A may not be a generator.
Output: $Q \in G_1 - \{\mathcal{O}\}$ satisfying $h_{q+1,A}(Q) = v$ if such Q exists. Otherwise, **nil**.
Procedure:

1: $u := v^{(q+1)/2}$;
2: if $u \notin \mathbf{F}_q$ then return **nil** ;
3: $x_1 := \xi(A) + u$; $x_2 := \xi(A) - u$;
4: Build a set $L_i := \{ Q \in E(\mathbf{F}_q) : \xi(Q) = x_i \}$ for $i = 1, 2$. // Note $0 \le {}^{\#}L_i \le 2$.
5: for each $Q \in L_1 \cup L_2$
6:    if $h_{q+1,A}(Q) = z$ then return Q ;
7: return **nil** ;

Before we evaluate computational complexity of our algorithm, we clarify assumptions on time complexities for operations on elements of $\mathbf{F}_q$ or $\mathbf{F}_r$. We assume that $\mathbf{F}_q$ and $\mathbf{F}_r$ are so realized that one arithmetic operation in $\mathbf{F}_q$ or $\mathbf{F}_r$ amounts to $O((\log q)^2)$ bit operations. We also assume that a generator g of 2-Sylow subgroup of $\mathbf{F}_q^{\times}$ is precomputed. This is achieved by a probabilistic algorithm which needs $O((\log q)^3)$ bit operations. Using g, we can deterministically compute a square root of a square element of $\mathbf{F}_q^{\times}$ with $O((\log q)^3)$ bit operations. We can also use g to construct $\mathbf{F}_r$ as $\mathbf{F}_q[T]/\langle T^2 - g \rangle$ where T is an indeterminate.

Theorem 2.2. *Algorithm 2.1 returns a correct result with $O((\log q)^3)$ bit operations.*

Proof. First, we prove correctness. Suppose there exists $Q \in G_1 - \{\mathcal{O}\}$ satisfying $h_{q+1,A}(Q) = v$. Recall that E is defined by the Weierstrass model. Since $A \in G_0 \subset E[q+1]$, we have

$$h_{q+1,A} = (\xi - \xi(A)) h_{q,A}. \tag{2.1}$$

Now key observation of our algorithm is $h_{q,A}(Q) \in \mu_l \subset \mu_{q+1}$ by Granger et al.[4, Theorem 2]. Thus evaluation of (2.1) at Q followed by $q+1$ powering yields

$$v^{q+1} = (\xi(Q) - \xi(A))^{q+1}. \tag{2.2}$$

Since $Q \in E(\mathbf{F}_q)-\{\mathcal{O}\}$, we have $\xi(Q) \in \mathbf{F}_q$. On the other hand $A \in G_0-\{\mathcal{O}\}$ implies $\xi(A)=\varphi_q(\xi(A))$. Thus $\xi(A) \in \mathbf{F}_q$. Therefore $\xi(Q)-\xi(A) \in \mathbf{F}_q$ and $(\xi(Q)-\xi(A))^{q+1}=(\xi(Q)-\xi(A))^2$. Substituting the right side of (2.2), we obtain

$$v^{q+1} = (\xi(Q)-\xi(A))^2. \tag{2.3}$$

Recall that q is odd. Hence

$$\xi(Q)-\xi(A) = \pm v^{(q+1)/2}.$$

Therefore $\xi(Q)$ is either x_1 or x_2. If $\xi(Q)=x_i$ then $Q \in L_i$ by the definition of L_i. Hence the algorithm terminates with correct output Q. This also implies that the algorithm reaches Step 7 only if there is no element Q in G_1 satisfying $h_{q+1,A}(Q)=v$.

Next, we evaluate computational complexity of Algorithm 2.1. Step 1 needs $O(\log q)$ multiplications in $\mathbf{F}_r$. For each i, we obtain L_i with $O(1)$ arithmetic operations and one square root computation in $\mathbf{F}_q$ (not in $\mathbf{F}_r$, which is ensured by Step 2). Since $G_0 \cap G_1=\{\mathcal{O}\}$, no division by zero occurs during evaluation of $h_{q+1,A}(Q)$ by the Miller algorithm. Hence we obtain $h_{q+1,A}(Q)$ for a given $Q \in G_1-\{\mathcal{O}\}$ with $O(\log q)$ arithmetic operations over $\mathbf{F}_r$. Thus the algorithm terminates with $O(\log q)$ arithmetic operations over $\mathbf{F}_r$ or $\mathbf{F}_q$ and at most two square root computations in $\mathbf{F}_q$. By our assumptions, they amount to $O((\log q)^3)$ bit operations. □

Remark 2.3. In case that q is a power of 2, the algorithm and its implementation are in fact easier because (2.3) yields a unique candidate of Q. However in cryptographic point of view, this case is irrelevant.

Example 2.4. Consider $E: Y^2=X^3-13X-7$ over $\mathbf{F}_{139}$ and take $l:=35$. Let θ be the class of T in $\mathbf{F}_{139}[T]/\langle T^2+4\rangle$. Then $\mathbf{F}_{139^2}=\mathbf{F}_{139}(\theta)$. Put $A:=(67, 38\theta)$ and $v:=25\theta+109$. Note that $\langle A\rangle=G_0$ and that v^{138} is a primitive 35-th root of unity. Then $u:=v^{70}=131$ and we obtain $x_1:=59$ and $x_2:=75$. Thus $L_1:=\{(59,\pm 54)\}$ and $L_2:=\{(75,\pm 1)\}$. The Miller algorithm gives $h_{140,A}((59, 54))=114\theta+109$, $h_{140,A}((59, -54))=25\theta+109$, $h_{140,A}((75, 1))=112\theta+22$ and $h_{140,A}((75, -1))=27\theta+22$. Therefore we obtain the desired answer $Q:=(59, -54)$.

We observe an example for a non-generator. Put $B:=5A$ and $z:=56\theta+55$ whose orders are both 7. There are five points $Q_n:=(83, 55)+n(69, 11) \in G_1$, where $0 \le n < 5$, satisfying $e_{140}(B, Q_n)=z$. Although the paring values are equal, the algorithm requires correct input from FEI, which are different for each n. For example, the algorithm returns unique point Q_0 for input $(4\theta+135, B)$, whereas it returns unique point Q_1 for input $(98\theta+41, B)$. It is a role of FEI to provide a correct value to Algorithm 2.1.

Acknowledgments. The author would like to thank Frederik Vercauteren and Steven Galbraith for their comments.

References

1. Bateman, P.T. and Horn, R.A.: A heuristic asymptotic formula concerning the distribution of prime numbers. Math. Comp., **16**, 363-367 (1962).
2. Duc, A. and Jetchev, D.: Hardness of computing individual bits for one-way functions on elliptic curves, Crypto 2012, Lect. Notes in Comput. Sci., **7417**, 832-849, ed. Safavi-Naini, R. and Canetti, R., Berlin, Heidelberg: Springer, 2012. doi: 10.1007/978-3-642-32009-5_48

3. Galbraith, S., Hess, F. and Vercauteren, F.: Aspects of Pairing inversion. IEEE Trans. Info. Theory, **54**, 5719-5728 (2008). doi: 10.1109/TIT.2008.2006431
4. Granger, R., Hess, F., Oyono, R., Thériault, N. and Vercauteren, F.: Ate pairing on hyperelliptic curves, Advances in Cryptology - EUROCRYPT 2007, Lect. Notes in Comput. Sci., **4515**, 430-447, ed. Naor, M., Springer, 2007. doi: 10.1007/978-3-540-72540-4_25
5. Karabina, K., Knapp, E. and Menezes, A.: Generalizations of Verheul's theorem to asymmetric pairings. Adv. Math. Comm., **7**, 103-111 (2013). doi: 10.3934/amc.2013.7.103
6. Lashermes, R., Fournier, J. and Goubin, L.: Inverting the final exponentiation of Tate pairings on ordinary elliptic curves using faults., CHES 2013, Lect. Notes in Comput. Sci., **8086**, 365-382, ed. Bertoni, G. and Coron, J.-S., Springer, 2013. doi: 10.1007/978-3-642-40349-1_21
7. Schoof, R.: Nonsingular plane cubic curves over finite fields. J. Combinatorial theory, Ser. A, **46**, 183-211 (1987). doi: 10.1016/0097-3165(87)90003-3
8. Vercauteren, F.: The hidden root problem, Paring-based cryptography 2008, Lect. Notes in Comput. Sci., **5209**, 89-99, Berlin, Heidelberg: Springer, 2008. doi: 10.1007/978-3-540-85538-5
9. Verheul, E.R.: Evidence that XTR is more secure than supersingular elliptic curve cryptosystems. J. Cryptology, **17**, 277-296 (2004). doi: 10.1007/s00145-004-0313-x