Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field

Alfred J. Menezes, Tatsuaki Okamoto, and Scott A. Vanstone

***Abstract*— Elliptic curve cryptosystems have the potential to provide relatively small block size, high-security public key schemes that can be efficiently implemented. As with other known public key schemes, such as RSA and discrete exponentiation in a finite field, some care must be exercised when selecting the parameters involved, in this case the elliptic curve and the underlying field. Specific classes of curves that give little or no advantage over previously known schemes are discussed. The main result of the paper is to demonstrate the reduction of the elliptic curve logarithm problem to the logarithm problem in the multiplicative group of an extension of the underlying finite field. For the class of supersingular elliptic curves, the reduction takes probabilistic polynomial time, thus providing a probabilistic subexponential time algorithm for the former problem.**

***Index Terms*—Discrete logarithms, elliptic curves, public key cryptography.**

I. Introduction

The DISCRETE LOGARITHM problem for a general group G can be stated as follows: given $\alpha \in G$ and $\beta \in G$, find an integer x such that $\beta = \alpha^x$, provided that such an integer exists. The integer x is called the *discrete logarithm* of β to the base α. In this paper, we shall consider the case where G is an elliptic curve group E defined over a finite field F_q, and where α is a point $P \in E(F_q)$.

In [1] and [2], Koblitz and Miller first proposed using the group of points on an elliptic curve over a finite field to construct public key cryptosystems. The security of these cryptosystems is based upon the presumed intractability of the problem of computing logarithms in the elliptic curve group. The best algorithms that are known for solving this problem are the exponential square root attacks (e.g., see [3]) that apply to any finite group and have a running time that is proportional to the square root of the largest prime factor dividing the order of the group. In [2], Miller argues that the index-calculus methods, which produced dramatic results in the computation of discrete logarithms in (the multiplicative group of) a finite field (see [3], [4]), do not extend to elliptic curve groups. Consequently, if the elliptic curve is chosen so that its order is divisible by a large prime, then even the best attacks take exponential time.

The method we propose in this paper reduces the elliptic curve logarithm problem in a curve E over a finite field F_q to the discrete logarithm problem in a suitable extension field F_{q^k} of F_q. This is achieved by establishing an isomorphism between $\langle P \rangle$, the subgroup of E generated by P, and the subgroup of nth roots of unity in F_{q^k}, where n denotes the order of P. The isomorphism is given by the Weil pairing.

Since the index-calculus methods for computing logarithms in a finite field have running times that are subexponential, the reduction is useful for the purpose of computing elliptic curve logarithms provided that k is small. This is indeed the case for special classes of elliptic curves, including many of the curves recommended for implementation in [1], [2], [5]–[7].

The remainder of the paper is organized as follows. In Section II, we list some of the properties of elliptic curves that we will use. In Section III, we describe the reduction, and in Section IV, we mention some special curves for which the reduction is especially useful. Finally, in Section V, we discuss some of the implications of our results for cryptography.

II. Background on Elliptic Curves

In this section, we review some of the theory of elliptic curves over finite fields which we will use. For further details, we refer the reader to the book by Silverman [8].

We use F_q to denote the finite field containing q elements, and denote the cyclic group of order n by $\mathbb{Z}_n$. Let E be an elliptic curve defined over F_q and let $q = p^m$, where p is the characteristic of F_q. If p is greater than 3, then $E(F_q)$ is the set of all solutions in $F_q \times F_q$ to an affine equation

$$y^2 = x^3 + ax + b, \tag{1}$$

with $a, b \in F_q$, $4a^3 + 27b^2 \neq 0$, together with an extra point $\mathcal{O}$, called the *point at infinity*. If $p = 2$, then an affine equation for $E(F_q)$ is

$$y^2 + a_3 y = x^3 + a_4 x + a_6, \tag{2}$$

with $a_3, a_4, a_6 \in F_q$, $a_3 \neq 0$, if the curve has j-invariant equal to 0, and

$$y^2 + xy = x^3 + a_2 x^2 + a_6, \tag{3}$$

with $a_2, a_6 \in F_q$, $a_6 \neq 0$, if the curve has j-invariant not equal to 0. There is a natural addition defined on the points of $E(F_q)$ that is given by the "tangent and chord method," and involves a few arithmetic operations in F_q. Under this addition,

Manuscript received June 8, 1991. This work was presented at the 23rd Annual ACM Symposium on Theory of Computing, New Orleans, LA, May 1991.

A. Menezes is with the Department of Discrete and Statistical Sciences, Auburn University, 120 Math Annex, Auburn, AL 36849.

T. Okamoto is with NTT Laboratories, 1-2356, Take, Yokosuka-shi, 238-03 Japan.

S. A. Vanstone is with the Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ON, Canada N2L 3G1.

IEEE Log Number 9210260.

0018-9448/93$03.00

the points of $E(F_q)$ form an abelian group of rank 1 or 2, with the point $\mathcal{O}$ serving as its identity element. By Hasse's theorem, the order of the group is $q+1-t$, where $|t| \leq 2\sqrt{q}$. The type of the group is (n_1, n_2), i.e., $E(F_q) \cong \mathbb{Z}_{n_1} \oplus \mathbb{Z}_{n_2}$, where $n_2 \mid n_1$, and furthermore $n_2 \mid q-1$. We will abuse the notation slightly, and call $E(F_q)$ an elliptic curve over F_q. The next result determines whether or not an elliptic curve of a certain order exists.

Lemma 1 ([9, (4.2)]): There exists an elliptic curve of order $q+1-t$ over F_q, if and only if one of the following conditions holds.

1) $t \not\equiv 0 \pmod{p}$ and $t^2 \leq 4q$.
2) m is odd and one of the following holds:
   a) $t = 0$;
   b) $t^2 = 2q$ and $p = 2$;
   c) $t^2 = 3q$ and $p = 3$.
3) m is even and one of the following holds:
   a) $t^2 = 4q$;
   b) $t^2 = q$ and $p \not\equiv 1 \pmod 3$;
   c) $t = 0$ and $p \not\equiv 1 \pmod 4$.

Let $\#E(F_q) = q+1-t$ denote the order of a curve. E is said to be *supersingular* if p divides t. From the preceding result, we can deduce that E is supersingular, if and only if $t^2 = 0$, q, $2q$, $3q$, or $4q$. The following result gives the group structure of the supersingular curves.

Lemma 2 ([9, 4.8)]): Let $\#E(F_q) = q+1-t$.

a) If $t^2 = q$, $2q$, or $3q$, then $E(F_q)$ is cyclic.
b) If $t^2 = 4q$, then either $E(F_q) \cong \mathbb{Z}_{\sqrt{q}-1} \oplus \mathbb{Z}_{\sqrt{q}-1}$ or $E(F_q) \cong \mathbb{Z}_{\sqrt{q}+1} \oplus \mathbb{Z}_{\sqrt{q}+1}$, depending on whether $t = 2\sqrt{q}$ or $t = -2\sqrt{q}$, respectively.
c) If $t = 0$ and $q \not\equiv 3 \pmod 4$, then $E(F_q)$ is cyclic. If $t = 0$ and $q \equiv 3 \pmod 4$, then either $E(F_q)$ is cyclic, or $E(F_q) \cong \mathbb{Z}_{(q+1)/2} \oplus \mathbb{Z}_2$.

The curve E can also be viewed as an elliptic curve over any extension field K of F_q; $E(F_q)$ is a subgroup of $E(K)$. The Weil theorem enables one to compute $\#E(F_{q^k})$ from $\#E(F_q)$ as follows. Let $t = q+1-\#E(F_q)$. Then $\#E(F_{q^k}) = q^k+1-\alpha^k-\beta^k$, where α, β are complex numbers determined from the factorization of $1-tT+qT^2 = (1-\alpha T)(1-\beta T)$.

An *n-torsion point* P is a point satisfying $nP = \mathcal{O}$. Let $E(F_q)[n]$ denote the subgroup of n-torsion points in $E(F_q)$, where $n \neq 0$. We will write $E[n]$ for $E(\overline{F}_q)[n]$, where $\overline{F}_q$ denotes the algebraic closure of F_q. If n and q are relatively prime, then $E[n] \cong \mathbb{Z}_n \oplus \mathbb{Z}_n$. If $n = p^e$, then either $E[p^e] \cong \{\mathcal{O}\}$ if E is supersingular, or else $E[p^e] \cong \mathbb{Z}_{p^e}$ if E is nonsupersingular.

The following result provides necessary and sufficient conditions for $E(F_q)$ to contain all of the n-torsion points in $E(\overline{F}_q)$. For definition of the terms in condition c), which we do not use in this paper, see [9].

Lemma 3 ([9, 3.7)]): If $\gcd(n, q) = 1$, then $E[n] \subset E(F_q)$, if and only if the following three conditions hold:

a) $n^2 \mid \#E(F_q)$;
b) $n \mid q-1$;
c) either $\phi \in \mathbb{Z}$ or $\vartheta(t^2 - 4q/n^2) \subset \mathrm{End}_{F_q}(E)$.

Let n be a positive integer relatively prime to q. The *Weil pairing* is a function

$$e_n\colon E[n] \times E[n] \to \overline{F}_q.$$

For a definition of the Weil pairing, see Appendix A. We list some useful properties of this function.

1) *Identity:* For all $P \in E[n]$, $e_n(P, P) = 1$.
2) *Alternation:* For all $P_1, P_2 \in E[n]$, $e_n(P_1, P_2) = e_n(P_2, P_1)^{-1}$.
3) *Bilinearity:* For all $P_1, P_2, P_3 \in E[n]$, $e_n(P_1 + P_2, P_3) = e_n(P_1, P_3)e_n(P_2, P_3)$, and $e_n(P_1, P_2 + P_3) = e_n(P_1, P_2)e_n(P_1, P_3)$.
4) *Nondegeneracy:* If $P_1 \in E[n]$, then $e_n(P_1, \mathcal{O}) = 1$. If $e_n(P_1, P_2) = 1$ for all $P_2 \in E[n]$, then $P_1 = \mathcal{O}$.
5) If $E[n] \subseteq E(F_{q^k})$, then $e_n(P_1, P_2) \in F_{q^k}$, for all $P_1, P_2 \in E[n]$.

Miller has developed an efficient probabilistic polynomial-time algorithm for computing the Weil pairing [10]. By a probabilistic polynomial algorithm, we mean a randomized algorithm whose expected running time is bounded by a polynomial in the size of the input. By a probabilistic subexponential algorithm with input x, we mean a randomized algorithm with expected running time bounded above by $L[\alpha, x]$, where $0 < \alpha < 1$ is a constant, and

$$L[\alpha, x] = \exp\left((c + o(1))(\ln x)^{\alpha}(\ln\ln x)^{1-\alpha}\right).$$

For a brief description of Miller's algorithm, see Appendix A. An implementation of the algorithm in MACSYMA, is given in [11].

The following result from [11] provides a method for partitioning the elements of an elliptic curve $E(F_q)$ into the cosets of $\langle P\rangle$, the subgroup of $E(F_q)$ generated by a point P of maximum order.

Lemma 4: Let $E(F_q)$ be an elliptic curve with group structure (n_1, n_2), and let P be an element of maximum order n_1. Then for all $P_1, P_2 \in E(F_q)$, P_1 and P_2 are in the same coset of $\langle P\rangle$, if and only if $e_{n_1}(P, P_1) = e_{n_1}(P, P_2)$.

The next result is similar to, and has a similar proof, as Lemma 4. For completeness, we include it here.

Lemma 5: Let $E(F_q)$ be an elliptic curve such that $E[n] \subseteq E(F_q)$, where n is a positive integer coprime to q. Let $P \in E[n]$ be a point of order n. Then, for all $P_1, P_2 \in E[n]$, P_1 and P_2 are in the same coset of $\langle P\rangle$ within $E[n]$, if and only if $e_n(P, P_1) = e_n(P, P_2)$.

Proof: If $P_1 = P_2 + kP$, then clearly

$$\begin{aligned} e_n(P, P_1) &= e_n(P, P_2)e_n(P, P)^k \\ &= e_n(P, P_2). \end{aligned}$$

Conversely, suppose that P_1 and P_2 are in different cosets of $\langle P\rangle$ within $E[n]$. Then we can write $P_1 - P_2 = a_1P + a_2Q$, where (P, Q) is a generating pair for $E[n] \cong \mathbb{Z}_n \oplus \mathbb{Z}_n$, and where $a_2Q \neq \mathcal{O}$. If $b_1P + b_2Q$ is any point in $E[n]$, then

$$\begin{aligned} e_n(a_2Q, b_1P + b_2Q) &= e_n(a_2Q, P)^{b_1}e_n(Q, Q)^{a_2b_2} \\ &= e_n(P, a_2Q)^{-b_1}. \end{aligned}$$

If $e_n(P, a_2Q) = 1$ then by the non-degeneracy property of e_n, we have that $a_2Q = \mathcal{O}$, a contradiction. Thus $e_n(P, a_2Q) \neq 1$. Finally,

$$\begin{aligned} e_n(P, P_1) &= e_n(P, P_2)e_n(P, P)^{a_1}e_n(P, a_2Q) \\ &\neq e_n(P, P_2). \end{aligned} \qquad \square$$

In the algorithms that follow, it is essential that we are able to pick points P uniformly and randomly on an elliptic curve $E(F_q)$ in probabilistic polynomial time. This can be accomplished as follows. We first randomly choose an element $x_1 \in F_q$. If x_1 is the x-coordinate of some point in $E(F_q)$, then we can find y_1 such that $(x_1, y_1) \in E(F_q)$ by solving a root finding problem in F_q. There are various techniques for finding the roots of a polynomial over F_q in probabilistic polynomial time; for example, see [12]. We then set $P = (x_1, y_1)$ or $(x_1, -y_1)$ if the curve has (1) (respectively, $P = (x_1, y_1)$ or $(x_1, y_1 + a_3)$, and $P = (x_1, y_1)$ or $(x_1, y_1 + x_1)$ if the curve has (2) or (3)). From Hasse's theorem, the probability that x_1 is the x-coordinate of some point in $E(F_q)$ is at least $1/2 - 1/\sqrt{q}$. Note that with the method just described the probability of picking a point of order 2 is twice the probability of picking any other point; this does not present a problem as there are at most three points of order 2.

Finally, for future reference, we state the following results.

Lemma 6: Let G be a group and $\alpha \in G$. Let $n = \prod_{i=1}^{k} p_i^{\beta_i}$ be the prime factorization of n. Then α has order n, if and only if

a) $\alpha^n = 1$, and
b) $\alpha^{n/p_i} \neq 1$ for each i, $1 \leq i \leq k$.

Lemma 7: Let G be an abelian group of type (cn, cn). If elements $\{\alpha_i\}$ are selected uniformly and randomly from G, then the elements $\{c\alpha_i\}$ are uniformly distributed about the elements of the subgroup of G of type (n, n).

III. The Reduction

Let $E(F_q)$ be an elliptic curve over the finite field F_q with group structure $\mathbb{Z}_{n_1} \oplus \mathbb{Z}_{n_2}$, where $n_2 \mid n_1$. Given the defining equation for $E(F_q)$, we can compute $\#E(F_q)$ in polynomial time by using Schoof's algorithm [13]. Also, we can determine n_1 and n_2 in probabilistic polynomial time by an algorithm due to Miller [10], given the integer factorization of $\gcd(\#E(F_q), q-1)$. We further assume that $\gcd(\#E(F_q), q) = 1$; it follows that $E[n_1] \cong \mathbb{Z}_{n_1} \oplus \mathbb{Z}_{n_1}$.

Let $P \in E(F_q)$ be a point of order n, where n divides n_1, and let $R \in E(F_q)$. We assume that n is known. The *elliptic curve logarithm problem* is the following: Given P and R, determine the unique integer l, $0 \leq l \leq n-1$, such that $R = lP$, provided that such an integer exists.

Since $e_n(P, P) = 1$, we deduce from Lemma 4 that $R \in \langle P \rangle$, if and only if $nR = \mathcal{O}$ and $e_n(P, R) = 1$, conditions which can be checked in probabilistic polynomial time. Henceforth, we will assume that $R \in \langle P \rangle$.

We first describe an algorithm for obtaining partial information about l by solving a discrete logarithm problem in the field F_q itself, in the case that P has maximum order.

Algorithm 1:

Input: An element $P \in E(F_q)$ of maximum order n_1, and $R = lP$.

Output: An integer $l' \equiv l \pmod{n'}$, where n' is a divisor of n_2.

1) Pick a random point $T \in E(F_q)$.
2) Compute $\alpha = e_{n_1}(P, T)$ and $\beta = e_{n_1}(R, T)$.
3) Compute l', the discrete logarithm of β to the base α in F_q.

Theorem 8: Algorithm 1 correctly computes $l' \equiv l \pmod{n'}$, where n' is some divisor of n_2.

Proof: Let $G \in E(F_q)$ be an element of order n_2 such that the pair of points (P, G) generates $E(F_q)$, and let $T = c_1P + c_2G$. Then,

$$\begin{aligned} \alpha^{n_2} &= e_{n_1}(P, T)^{n_2} = e_{n_1}(P, P)^{c_1n_2}e_{n_1}(P, c_2n_2G) \\ &= e_{n_1}(P, \mathcal{O}) = 1, \end{aligned}$$

and hence the order of α, denoted n', divides n_2. Since $n_2 \mid q-1$ it also follows that $\alpha \in F_q$. Now, since

$$\beta = e_{n_1}(R, T) = e_{n_1}(lP, T) = e_{n_1}(P, T)^l = \alpha^l = \alpha^{l'},$$

we can then determine l' by computing the discrete logarithm of β to the base α in F_q. $\square$

Since there are n_2 cosets of $\langle P \rangle$ within $E(F_q)$, we deduce from Lemma 4 that the probability that $n' = n_2$ is $\phi(n_2)/n_2$. If n_2 is small compared to n_1 however (as is expected if the curve is randomly chosen since $n_2 \mid \gcd(n_1, q-1)$), then this method does not provide us with any significant information about l. In the remainder of this section, we describe a technique for computing l modulo n.

Let k be the smallest positive integer such that $E[n] \subseteq E(F_{q^k})$; it is clear that such an integer k exists.

Theorem 9: There exists $Q \in E[n]$, such that $e_n(P, Q)$ is a primitive nth root of unity.

Proof: Let $Q \in E[n]$. Then, by the bilinearity of the Weil pairing, we have that

$$e_n(P, Q)^n = e_n(P, nQ) = e_n(P, \mathcal{O}) = 1.$$

Thus $e_n(P, Q) \in \mu_n$, where μ_n denotes the subgroup of the nth roots of unity in F_{q^k}. There are n cosets of $\langle P \rangle$ within $E[n]$, and by Lemma 5 we deduce that as Q varies among the representatives of these n cosets, $e_n(P, Q)$ varies among all of the elements of μ_n. The result now follows. $\square$

Let Q be a point in $E[n]$ such that $e_n(P, Q)$ is a primitive nth root of unity. The proof of the next result is straightforward.

Theorem 10: Let $f\colon \langle P \rangle \to \mu_n$ be defined by $f\colon R \mapsto e_n(R, Q)$. Then f is a group isomorphism.

We can now describe the method for reducing the elliptic curve logarithm problem to the discrete logarithm problem in a finite field.

TABLE I
SOME INFORMATION ABOUT SUPERSINGULAR CURVES

Class of curve	t	Group structure	n_1	k	Type of $E(F_{q^k})$	c
I	0	cyclic	$q+1$	2	$(q+1, q+1)$	1
II	0	$\mathbf{Z}_{(q+1)/2} \oplus \mathbf{Z}_2$	$(q+1)/2$	2	$(q+1, q+1)$	2
III	$\pm\sqrt{q}$	cyclic	$q+1 \mp \sqrt{q}$	3	$(\sqrt{q^3} \pm 1, \sqrt{q^3} \pm 1)$	$\sqrt{q} \pm 1$
IV	$\pm\sqrt{2q}$	cyclic	$q+1 \mp \sqrt{2q}$	4	(q^2+1, q^2+1)	$q \pm \sqrt{2q}+1$
V	$\pm\sqrt{3q}$	cyclic	$q+1 \mp \sqrt{3q}$	6	(q^3+1, q^3+1)	$(q+1)(q \pm \sqrt{3q}+1)$
VI	$\pm 2\sqrt{q}$	$\mathbf{Z}_{\sqrt{q}\mp 1} \oplus \mathbf{Z}_{\sqrt{q}\mp 1}$	$\sqrt{q} \mp 1$	1	$(\sqrt{q} \mp 1, \sqrt{q} \mp 1)$	1

Algorithm 2:
Input: An element $P \in E(F_q)$ of order n, and $R \in \langle P \rangle$.
Output: An integer l such that $R = lP$.

1) Determine the smallest integer k such that $E[n] \subseteq E(F_{q^k})$.
2) Find $Q \in E[n]$ such that $\alpha = e_n(P, Q)$ has order n.
3) Compute $\beta = e_n(R, Q)$.
4) Compute l, the discrete logarithm of β to the base α in F_{q^k}.

Note that the output of Algorithm 2 is correct since

$$\beta = e_n(lP, Q) = e_n(P, Q)^l = \alpha^l.$$

Remarks: The reduction described in this section takes exponential time (in $\ln q$) in general, as k is exponentially large in general. Algorithm 2 is also incomplete as we have not provided methods for determining k, and for finding the point Q. We shall accomplish this in the next section for the supersingular elliptic curves.

IV. SUPERSINGULAR CURVES

In this section, we prove that the reduction of Algorithm 2 takes probabilistic polynomial time for supersingular curves, resulting in a probabilistic subexponential time algorithm for computing elliptic curve logarithms in these curves.

Let $E(F_q)$ be a supersingular elliptic curve of order $q+1-t$ over F_q, and let $q = p^m$. By Lemmas 1 and 2, E lies in one of the following classes of curves.

(I) $t = 0$ and $E(F_q) \cong \mathbf{Z}_{q+1}$.
(II) $t = 0$ and $E(F_q) \cong \mathbf{Z}_{(q+1)/2} \oplus \mathbf{Z}_2$ (and $q \equiv 3 \pmod 4$).
(III) $t^2 = q$ (and m is even).
(IV) $t^2 = 2q$ (and $p = 2$ and m is odd).
(V) $t^2 = 3q$ (and $p = 3$ and m is odd).
(VI) $t^2 = 4q$ (and m is even).

Let P be a point of order n in $E(F_q)$. Since $n_1 \mid (q+1-t)$, and $p \mid t$, we have $\gcd(n_1, q) = 1$. By applying the Weil theorem and using Lemma 2, one can easily determine the smallest positive integer k such that $E[n_1] \subseteq E(F_{q^k})$, and hence $E[n] \subseteq (F_{q^k})$. For convenience, we summarize the relevant information in Table I. Note that for each class of curves, the structure of $E(F_{q^k})$ is of the form $\mathbf{Z}_{cn_1} \oplus \mathbf{Z}_{cn_1}$, for appropriate c. We now proceed to give a detailed description of the reduction for supersingular curves.

Algorithm 3:
Input: An element P of order n on a supersingular curve $E(F_q)$, and $R \in \langle P \rangle$.
Output: An integer l such that $R = lP$.

1) Determine k and c from Table I.
2) Pick a random point $Q' \in E(F_{q^k})$ and SET $Q = (cn_1/n)Q'$.
3) Compute $\alpha = e_n(P, Q)$ and $\beta = e_n(R, Q)$.
4) Compute the discrete logarithm l' of β to the base α in F_{q^k}.
5) Check whether $l'P = R$. IF this is so, THEN $l = l'$ and we are done. Otherwise, the order of α must be less than n, so GO TO 2).

Note that by Lemma 7, Q is a random point in $E[n]$. Note also that the probability that the field element α has order n is $\phi(n)/n$. This follows from Lemma 5 and the facts that there are $\phi(n)$ elements of order n in F_{q^k}, and there are n cosets of $\langle P \rangle$ within $E[n]$.

We now proceed to prove the main result of this section.

Theorem 11: If $E(F_q)$ is a supersingular curve, then the reduction of the elliptic curve logarithm problem in $E(F_q)$ to the discrete logarithm problem in F_{q^k} is a probabilistic polynomial time (in $\ln q$) reduction.

Proof: We assume that a basis of the field F_q over its prime field is explicitly given. To do arithmetic in F_{q^k}, we need to find an irreducible polynomial $f(x)$ of degree k over F_q. This can be done in probabilistic polynomial time, for example by the method given in [12]. We then have $F_{q^k} \cong F_q[x]/I_f$, where I_f denotes the ideal generated by $f(x)$. Note that the constant polynomials in $F_q[x]$ form a subfield isomorphic to F_q. The point Q' can be chosen in probabilistic polynomial time since $Q' \in E(F_{q^k})$ and $k \le 6$, and then Q can be determined in polynomial time. The elements α and β can be computed in probabilistic polynomial time by Miller's algorithm. Since

$$\frac{n}{\phi(n)} \le 6 \ln\ln n, \qquad \text{for } n \ge 5,$$

(see [14]), the expected number of iterations before we find a Q such that $e_n(P, Q)$ has order n is $O(\ln\ln n)$. Finally, observe that $l'P = R$ can be tested in polynomial time, and that $n = O(q)$. □

We note that it is unknown whether there exist subexponential algorithms for the discrete logarithm problem in fields F_{q^k} as both q and k tend to infinity. Subexponential algorithms with rigorously proved expected running times of $L[1/2, q]$ are known for the cases $q = 2$ [15], $q = p$ and $k = 1$ [15], $q = p$ and $k = 2$ [16], and $q = p$ and $\log p < n^{0.98}$ [16]. Practical subexponential algorithms with heuristic expected running times of $L[1/3, q]$ are known for the cases $q = 2$ [4] or q a fixed prime [17], and with heuristic expected running times of $L[1/2, q]$ for the cases $q = p$ and $k = 1$ [18], and $q = p$ and k fixed [19] (the latter is described for the case $k = 2$ but applies to the case k fixed [3]). Algorithms with heuristic expected running times of $L[1/3, q]$ are known for the cases $q = p$ and $k = 1$ [20], and $q = p$ and k fixed [21], however, these do not appear practical at present.

Note that the discrete logarithm problem in F_{q^k} solved in step 4) of Algorithm 3 has a base element α of order n, where $n < q^k - 1$. The probabilistic subexponential algorithms mentioned above for computing discrete logarithms in a finite field require that the base element be primitive. Using these algorithms, we obtain the following.

Corollary 12: Let P be an element of order n in a supersingular elliptic curve $E(F_q)$, and let $R = lP$ be a point in $E(F_q)$. If q is a prime, or if q is a prime power $q = p^m$, where p is fixed, then the new algorithm can determine l in probabilistic subexponential time.

Proof: The problem of finding the logarithm of β to the case α in F_{q^k} can be solved in probabilistic subexponential time as follows. We first obtain the integer factorization of $q^k - 1$ in probabilistic subexponential time using one of the many techniques available for integer factorization (for example [22] or [23] for practical algorithms with heuristic running time analyses, and [15] for an algorithm with a rigorous running time analysis). Observe that we *a priori* have the following partial factorizations of $q^k - 1$.

(I) $q^2 - 1 = (q + 1)(q - 1)$.
(II) $q^3 - 1 = (q - 1)(q + 1 - \sqrt{q})(q + 1 + \sqrt{q})$.
(III) $q^4 - 1 = (q - 1)(q + 1)(q + 1 - \sqrt{2q})(q + 1 + \sqrt{2q})$.
(IV) $q^6 - 1 = (q-1)(q+1)(q+1-\sqrt{3q})(q+1+\sqrt{3q})(q^2 + q + 1)$.

We then select random elements γ in F_{q^k}, until γ has order $q^k - 1$; the expected number of trials is $(q^k - 1)/\phi(q^k - 1)$ which is $O(\ln\ln q)$ since $k \leq 6$. The order of γ can be checked in polynomial time using Lemma 6. By solving two discrete logarithm problems in F_{q^k}, we find the unique integers s and t, $0 \leq s, t \leq q^k - 1$, such that $\alpha = \gamma^s$ and $\beta = \gamma^t$. Since $\beta = \alpha^{l'}$, we obtain the congruence $sl' \equiv t \pmod{q^k - 1}$. Let $w = \gcd(s, q^k - 1)$, and let $v = (q^k - 1)/w$ be the order of α. Then $l' = (s/w)^{-1}(t/w) \pmod{v}$.

The logarithms in F_{q^k} can be computed in probabilistic subexponential time in $\ln q^k$ (and consequently also subexponential in $\ln q$) using, for example, the algorithm in [18] if q is prime and $k = 1$, [19] if q is prime and $k > 1$, or [4], [17] if q is the power of a fixed prime. □

In solving the elliptic curve logarithm problem in practice, one would first factor n. Using this factorization, we can easily check the order of α. Thus to find Q, we repeatedly choose random points in $E[n]$ until α has order n. This avoids the possibility of having to solve several discrete logarithm problems before l' is in fact equal to l. Note however that this modified reduction is different from the reduction described in Algorithm 3, and in particular is no longer a probabilistic polynomial time reduction to the discrete logarithm problem in a finite field.

The dominant steps of the algorithm as modified in the previous paragraph are the factoring of $q^k - 1$ and in the final stage of computing discrete logarithms in F_{q^k}. The number field sieve [23] for factoring an integer n has an expected running time of $L[1/3, n]$. The expected running time of the algorithm is thus either $L[1/2, q^k]$ or $L[1/3, q^k]$ depending on the running time of the best algorithm known for the discrete logarithm problem in F_{q^k}.

We conclude that for the supersingular curves, the elliptic curve discrete logarithm problem is more tractable than previously believed. Among these special elliptic curves are the following.

(A) $y^2 + y = x^3 + b$ over F_{2^m}, m odd (class I).
(B) $y^2 = x^3 - ax$ over F_p, where $p > 3$ is a prime, a is a quadratic nonresidue in F_p, and $p \equiv 3 \pmod 4$ (class I).
(C) $y^2 = x^3 - ax$ over F_p, where $p > 3$ is a prime, a is a quadratic residue in F_p, and $p \equiv 3 \pmod 4$ (class II).
(D) $y^2 = x^3 + b$ over F_p, where $p > 3$ is a prime, and $p \equiv 2 \pmod 3$ (class I).

We will discuss these curves further in the next section.

V. Cryptographic Implications

In order to implement the Diffie–Hellman and El Gamal protocols [1], one would like a cyclic group which is relatively easy to exponentiate in, and one for which the discrete logarithm problem is intractable.

Elliptic curve cryptosystems have the potential to be implemented efficiently with relatively small block size, and high security. (This was, of course, the motivation for studying such systems.) With current schemes, such as RSA and discrete exponentiation in a finite field, block sizes in excess of 500 bits (and preferably 1,000 bits) are necessary for adequate security. The results of the preceding section show that some care must be exercised in selecting an elliptic curve over a finite field. This is not unlike the situation with RSA where the prime numbers must be judiciously chosen. It is now clear that the curve

$$y^2 + y = x^3$$

over F_{2^m} is no more secure than using the cyclic group of nonzero elements in $F_{2^{2m}}$. Since it appears that the cost of computations on the curve is higher than the cost of computations in $F_{2^{2m}}$, such a curve is inferior for cryptographic purposes to other existing systems. Similar statements are valid for the classes of curves (B), (C), and (D) of Section IV.

The curve $y^2 + y = x^3$ over F_{2^m} was first considered for the implementation of elliptic curve cryptosystems by Koblitz [1]. In [6], the authors suggested the particular values $m = 61$ and $m = 127$. Since the discrete logarithm problem in the fields $F_{2^{122}}$ and $F_{2^{254}}$ are very tractable using the index-calculus methods (see [24]), these curves are clearly inadequate for

TABLE II
SOME USEFUL SUPERSINGULAR CURVES OVER F_{2^m}

m	Curve	Order of curve over F_{2^m}	Rough estimate of the operation count for an index-calculus attack in $F_{2^{4m}}$
173	E_1	$5 \cdot 13625405957 \cdot$ P42	1.4×10^{18}
173	E_2	$7152893721041 \cdot$ P40	1.4×10^{18}
179	E_2	$1301260549 \cdot$ P45	2.5×10^{18}
191	E_1	$5 \cdot 3821 \cdot 89618875387061 \cdot$ P40	8.6×10^{18}
191	E_2	$25212001 \cdot 5972216269 \cdot$ P41	8.6×10^{18}
233	E_1	$5 \cdot 3108221 \cdot$ P63	4.3×10^{20}
239	E_1	$5 \cdot 77852679293 \cdot$ P61	7.2×10^{20}
239	E_2	P72	7.2×10^{20}
281	E_2	$91568909 \cdot$ PRP77	2.3×10^{22}
323	E_2	$137 \cdot 953 \cdot 525313 \cdot$ P87	5.3×10^{23}

cryptographic purposes. The particular values $m = 191$ and $m = 251$ were suggested in [7]. These curves should also be avoided for the same reasons. The class of curves (B) and (C) were suggested by Miller [2]. Finally, the class of curves (D) was suggested in [6] for the implementation of elliptic curve cryptosystems, and by Kaliski [5] for the implementation of secure pseudorandom number generators.

The following cyclic curves over F_{2^m} (m odd)

$$E_1\colon\ y^2 + y = x^3 + x$$

and

$$E_2\colon\ y^2 + y = x^3 + x + 1$$

are much more attractive since they are easily implementable (see [25]), and give a security level that is apparently equivalent to the multiplicative group of $F_{2^{4m}}$ (see Lemma 13 in Appendix B). In Table II, we list several values of m, m odd, for which the order of either the curve E_1 or E_2 contains a large prime factor, precluding a square-root attack. The factorizations of $\#E_1$ and $\#E_2$ were obtained from [26]. The approximate running time for an index calculus attack in $F_{2^{4m}}$ is also included, using the asymptotic running time estimate of

$$\exp\left((1.35)n^{1/3}(\ln n)^{2/3}\right)$$

operations for computing discrete logarithms in F_{2^n} [3].

It should be noted that although the supersingular curves over F_{2^m} have received the most attention to date, this does not mean that the more general class of curves is unattractive for implementation. If a nonsupersingular curve is desired, then the attack of Section III can be avoided by simply choosing a nonsupersingular curve $E(F_q)$ such that the corresponding k value is sufficiently large so that the discrete logarithm problem in F_{q^k} is considered intractable. Let $E(F_q)$ have type (n_1, n_2). Let P be a point of order n and assume that n is divisible by a large prime v. To ensure that $k \neq l$, we must check that either v does not divide $q^l - 1$ or that v^2 does not divide $\#E(F_{q^l})$. The quantity $\#E(F_{q^l})$ can be easily obtained from $\#E(F_q)$ by applying the Weil theorem, as described in Section II. Some work has been done on the implementation of nonsupersingular curves over F_{2^m} and this is reported in [25].

APPENDIX A
WEIL PAIRING

We give a brief introduction to the theory of divisors, define the Weil pairing, and outline Miller's algorithm for computing the Weil pairing. For a more thorough treatment, we refer to [8] and [10].

Let $K = F_q$ and let $\overline{K}$ denote its algebraic closure. Let E be an elliptic curve defined over K. If L is any field containing K, then $E(L)$ denotes the set of points on the curve whose coordinates are both in L, and including the point at infinity. We will write E for $E(\overline{K})$.

A *divisor* D is a formal sum of points in E, $D = \sum_{P \in E} n_P(P)$, where $n_P \in \mathbb{Z}$, and $n_P = 0$ for all but finitely many $P \in E$. The *degree* of D is the integer $\sum n_P$. The divisors of degree 0 form an additive group, denoted D^0. The *support* of D is the set $\{P \in E \mid n_P \neq 0\}$.

If E is defined by the equation $r(x, y) = 0$, $r \in K[x, y]$, then the *function field K(E)* of E over K is the field of fractions of the domain $K[x, y]/I_r$, where I_r denotes the ideal generated by r. Similarly, $\overline{K}(E)$ is the field of fractions of $\overline{K}[x, y]/I_r$.

Let $f \in \overline{K}(E)^*$. For each $P \in E$, define $v_P(f)$ to be $n > 0$ or $-n < 0$ if f has a zero or a pole of order n at P, respectively. We associate the divisor $\sum v_P(f)(P)$ to f, and denote it by (f). One can verify that $(f) \in D^0$. A divisor $D = \sum n_P(P)$ is said to be *principal* if $D = (f)$ for some $f \in \overline{K}(E)^*$. One can also verify that D is principal, if and only if $\sum n_P = 0$ and $\sum n_P P = \mathcal{O}$.

Let D_l denote the set of all principal divisors; D_l forms a subgroup of D^0. If $D_1, D_2 \in D^0$, we write $D_1 \sim D_2$ if $D_1 - D_2 \in D_l$. For each $D \in D^0$ there exists a unique point $P \in E$ such that $D \sim (P) - (\mathcal{O})$.

If $D = \sum n_P(P)$ is a divisor and $f \in \overline{K}(E)^*$ such that D and (f) have disjoint supports, then we define $f(D) = \prod_{P \in E} f(P)^{n_P}$.

Now, let m be an integer coprime to q and let $P, Q \in E[m]$. Let $A, B \in D^0$ such that $A \sim (P) - (\mathcal{O})$ and $B \sim (Q) - (\mathcal{O})$, and A and B have disjoint supports. Let $f_A, f_B \in \overline{K}(E)$ be such that $(f_A) = mA$ and $(f_B) = mB$. Then the Weil pairing

$e_m(P, Q)$ is defined to be

$$e_m(P, Q) = \frac{f_A(B)}{f_B(A)}.$$

We now briefly outline Miller's algorithm [10] for computing the Weil pairing. Let $D_1, D_2 \in D^0$ with $D_1 = (P_1) - (\mathcal{O}) + (f_1)$, $D_2 = (P_2) - (\mathcal{O}) + (f_2)$, where $P_1, P_2 \in E$ and $f_1, f_2 \in \overline{K}(E)$. Then $D_1 + D_2 = (P_3) - (\mathcal{O}) + (f_1 f_2 f_3)$, where $P_3 = P_1 + P_2$, and $f_3 = l/v$, where l is the equation of the line through P_1 and P_2, and v is the equation of the vertical line through P_3.

If $D = \sum n_P(P)$ is a principal divisor, then we can find $f \in \overline{K}(E)$ such that $D = (f)$ by first writing $D = \sum n_P((P) - (\mathcal{O}))$, and then repeatedly using the method of the previous paragraph to compute each term of the sum. Note that if $P \in E(K)$ for each P in the support of D, then $f \in K(E)$, and all computations take place in the field K itself. The problem with this method is that the function f may itself be of exponential size, relative to the size of the input D. Hence, instead of writing f explicitly, i.e., writing down all of the nonzero coefficients and the corresponding monomials of f, we keep f in factored form. The factored form will be of polynomial size, and f can be evaluated at points P in polynomial time (provided that $f(P)$ is defined). As a result of the method of this construction, f may be undefined at most on all points of the supports of the divisors occurring in the intermediate steps.

To find f_A and f_B in order to compute $e_m(P, Q)$, we first fix an addition chain $1 = a_1, a_2, \cdots, a_t = m$, where $t \leq 2\log_2 m$. We pick random points $T, U \in E(K)$ such that $P + T$ and T are distinct from $\pm a_i U$ and $\pm a_i(Q + U)$, and $Q + U$ and U are distinct from $\pm a_i T$ and $\pm a_i(P + T)$, for each i, $1 \leq i \leq t$. Let $A = (P + T) - (T)$, $B = (Q + U) - (U)$. We then compute f_A and f_B by the method previously described. Finally, we compute

$$e_m(P, Q) = \frac{f_A(Q + U) f_B(T)}{f_B(P + T) f_A(U)},$$

and this is defined by choice of T and U.

The number of pairs of points $(T, U) \in E(K) \times E(K)$ that do not satisfy the previous conditions is at most $16t\#E(K)$. Thus, if $m \approx \#E(K)$ and $m \geq 1024$, then the probability of success is at least 1/2. Consequently, the algorithm to compute $e_m(P, Q)$ is a probabilistic polynomial time algorithm.

Appendix B

We show that $k = 4$ for the curves E_1 and E_2 considered in Section V. There are precisely 3 isomorphism classes of supersingular elliptic curves over F_{2^m}, when m is odd. A representative from each class is given next.

$$E_1\colon\ y^2 + y = x^3 + x \tag{4}$$

$$E_2\colon\ y^2 + y = x^3 + x + 1 \tag{5}$$

$$E_3\colon\ y^2 + y = x^3. \tag{6}$$

The order of these curves is listed next. We let $q = 2^m$:

$$\#E_1(F_{2^m}) = \begin{cases} q + 1 - 2\sqrt{q} & m \equiv 0 \pmod 8 \\ q + 1 + \sqrt{2q} & m \equiv 1, 7 \pmod 8 \\ q + 1 & m \equiv 2, 6 \pmod 8 \\ q + 1 - \sqrt{2q} & m \equiv 3, 5 \pmod 8 \\ q + 1 + 2\sqrt{q} & m \equiv 4 \pmod 8; \end{cases}$$

$$\#E_2(F_{2^m}) = \begin{cases} q + 1 + 2\sqrt{q} & m \equiv 0 \pmod 8 \\ q + 1 - \sqrt{2q} & m \equiv 1, 7 \pmod 8 \\ q + 1 & m \equiv 2, 6 \pmod 8 \\ q + 1 + \sqrt{2q} & m \equiv 3, 5 \pmod 8 \\ q + 1 - 2\sqrt{q} & m \equiv 4 \pmod 8; \end{cases}$$

$$\#E_3(F_{2^m}) = \begin{cases} q + 1 - 2\sqrt{q} & m \equiv 0 \pmod 4 \\ q + 1 & m \equiv 1, 3 \pmod 4 \\ q + 1 + 2\sqrt{q} & m \equiv 2 \pmod 4. \end{cases}$$

Lemma 13: For the curves $E_1(F_{2^m})$ and $E_2(F_{2^m})$, m odd, we have $k = 4$.

Proof: We prove the result for E_1 when $m \equiv 1$ or 7 (mod 8). The remaining cases are dealt with in a similar fashion. Henceforth we assume that $m \equiv 1$ or 7 (mod 8). Let $q = 2^m$ and $n = \#E_1(F_q)$. By Lemma 2a), $E_1(F_q)$ is cyclic. Now, $\#E_1(F_{q^2}) = q^2 + 1$ and $\#E_1(F_{q^3}) = q^3 + 1 - \sqrt{2q^3}$. By Lemma 2c), $E_1(F_{q^2})$ is cyclic, and by Lemma 2a), $E_1(F_{q^3})$ is cyclic. Consequently

$$E_1(F_{q^2}) \bigcap E_1[n] = E_1(F_q),$$

and

$$E_1(F_{q^3}) \bigcap E_1[n] = E_1(F_q).$$

Finally, $\#E_1(F_{q^4}) = q^4 + 1 + 2\sqrt{q^4}$, and by Lemma 2b) we have that $E_1(F_{q^4}) \cong \mathbb{Z}_{(q^2+1)} \oplus \mathbb{Z}_{(q^2+1)}$. Since

$$q^2 + 1 = (q + 1 + \sqrt{2q})(q + 1 - \sqrt{2q}),$$

it follows that $E_1[n] \subseteq E_1(F_{q^4})$. □

References

[1] N. Koblitz, "Elliptic curve cryptosystems," *Math. Computat.*, vol. 48, pp. 203–209, 1987.

[2] V. Miller, "Uses of elliptic curves in cryptography," in *Advances in Cryptology—Crypto '85* (Lecture Notes in Computer Sciences), vol. 218. New York: Springer-Verlag, 1986, pp. 417–426.

[3] A. Odlyzko, "Discrete logarithms and their cryptographic significance," in *Advances in Cryptology—Eurocrypt '84* (Lecture Notes in Computer Science), vol. 209. New York: Springer-Verlag, 1985, pp. 224–314.

[4] D. Coppersmith, "Fast evaluation of logarithms in fields of characteristic two," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 587–594, 1984.

[5] B. Kaliski, "A pseudorandom bit generator based on elliptic logarithms," in *Advances in Cryptology—Crypto '86* (Lecture Notes in Computer Science), vol. 293. New York: Springer-Verlag, 1987, pp. 84–103.

[6] A. Bender and G. Castagnoli, "On the implementation of elliptic curve cryptosystems," in *Advances in Cryptology—Crypto '89* (Lecture Notes in Computer Science), vol. 435. New York: Springer-Verlag, 1990, pp. 417–426.

[7] A. Menezes and S. Vanstone, "The implementation of elliptic curve cryptosystems," in *Advances in Cryptology—Auscrypt '90* (Lecture Notes in Computer Science), vol. 453. New York: Springer-Verlag, 1990, pp. 2–13.

[8] J. Silverman, *The Arithmetic of Elliptic Curves.* New York: Springer-Verlag, 1986.

[9] R. Schoof, "Nonsingular plane cubic curves over finite fields," *J. Combinat. Theory*, vol. A 46, pp. 183–211, 1987.
[10] V. Miller, "Short programs for functions on curves," unpublished manuscript, 1986.
[11] B. Kaliski, "Elliptic curves and cryptography: A pseudorandom bit generator and other tools," Ph.D. thesis, M.I.T., Jan. 1988.
[12] M. Ben-Or, "Probabilistic algorithms in finite fields," in *Proc. 22nd IEEE Symp. Foundations Comput. Sci.*, 1981, pp. 394–398.
[13] R. Schoof, "Elliptic curves over finite fields and the computation of square roots mod p," *Math. Computat.*, vol. 44, pp. 483–494, 1985.
[14] J. Rosser and L. Schoenfield, "Approximate formulas for some functions of prime numbers," *Illinois J. Math.*, vol. 6, pp. 64–94, 1962.
[15] C. Pomerance, "Fast, rigorous factorization and discrete logarithms algorithms," in *Discrete Algorithms and Complexity.* New York: Academic Press, 1987, pp. 119–143.
[16] R. Lovorn, "Rigorous, subexponential algorithms for discrete logarithms over finite fields," Ph.D. thesis, Univ. of Georgia, 1992.
[17] M. Hellman and M. Reyneri, "Fast computation of discrete logarithms in GF(q)," in *Advances in Cryptology—Crypto '82.* New York: Plenum Press, 1983, pp. 3–13.
[18] D. Coppersmith, A. Odlyzko, and R. Schroeppel, "Discrete logarithms in GF(p)," *Algorithmica*, vol. 1, pp. 1–15, 1986.
[19] T. ElGamal, "A subexponential-time algorithm for computing discrete logarithms over GF(p^2)," *IEEE Trans. Inform. Theory*, vol. IT-31, pp. 473–481, 1985.
[20] D. Gordon, "Discrete logarithms in GF(p) using the number field sieve," *SIAM J. Disc. Math.*, vol. 6, pp. 124–138, 1993.
[21] ——, "Discrete logarithms in GF(p^n) using the number field sieve," preprint, 1991.
[22] R. Silverman, "The multiple polynomial quadratic sieve," *Math. Computat.*, vol. 48, pp. 329–339, 1987.
[23] A. Lenstra, H. W. Lenstra, M. Manasse, and J. Pollard, "The number field sieve," in *Proc. 22nd ACM Symp. Theory of Computing*, 1990, pp. 564–572.
[24] D. Gordon and K. McCurley, "Massively parallel computation of discrete logarithms," in *Advances in Cryptology—Crypto '92*, to appear.
[25] A. Menezes and S. Vanstone, "Elliptic curve cryptosystems and their implementation," *J. Cryptology,* vol. 6, 1993.
[26] J. Brillhart, D. Lehmer, J. Selfridge, B. Tuckerman and S. Wagstaff, "Factorizations of $b^n \pm 1$ up to high powers," *Contemporary Math.* (Amer. Math. Soc.), vol. 22, 1983.