Evaluation of Security Level of Cryptography: The Elliptic Curve Discrete Logarithm Problem (ECDLP)

Alfred Menezes
University of Waterloo
Contact: ajmeneze@uwaterloo.ca

December 14, 2001

Contents

1 Executive Summary

The hardness of the elliptic curve discrete logarithm problem (ECDLP) is crucial for the security of elliptic curve cryptographic schemes. This report describes the state-of-the-art in algorithms for solving the ECDLP. The special classes of elliptic curves that are known to be weak for cryptographic purposes are identified, and methods for avoiding these weak elliptic curves are described.

2 Introduction

In 1985, Neal Koblitz [35] and Victor Miller [47] independently proposed using the group of points on an elliptic curve defined over a finite field in discrete logarithm cryptographic systems. The primary advantage that elliptic curve systems have over systems based on the multiplicative group of a finite field (and also over systems based on the intractability of integer factorization) is the absence of a subexponential-time algorithm (such as those of "index-calculus" type) that could find discrete logarithms in these groups. Consequently, one can use an elliptic curve group that is smaller in size while maintaining the same level of security. The result is smaller key sizes, bandwidth savings, and faster implementations—features which are especially attractive for security applications where computational power and integrated circuit space is limited, such as smart cards, personal digital assistants, and wireless devices.

Elliptic curve cryptographic protocols for digital signatures, public-key encryption, and key establishment have been standardized by numerous standards organizations including:

1. American National Standards Institute (ANSI X9.62 [2], ANSI X9.63 [3]).
2. Institute of Electrical and Electronics Engineers (IEEE 1363-2000 [30]).
3. International Standards Organization (ISO/IEC 15946-3 [31]).
4. U.S. government's National Institute for Standards and Technology (FIPS 186-2 [50]).
5. Wireless Application Protocol Forum (WAP WTLS [71]).
6. Internet Engineering Task Force (IETF PKIX [7], IETF OAKLEY [32]).
7. Standards for Efficient Cryptography Group (SECG [69]).

2.1 The ECDLP

A necessary condition for the security of all elliptic curve cryptographic schemes is that the *elliptic curve discrete logarithm problem* (ECDLP) be intractable. In this problem, one is given an elliptic curve E defined over a finite field $\mathbb{F}_q$, a point P of order n on E, and a point Q that is a multiple of P, and one has to find the integer $l \in [0, n-1]$ such that $Q = lP$.

In elliptic curve cryptographic schemes, the non-secret parameters E, $\mathbb{F}_q$, P and n are first chosen. Then, an entity selects an integer d uniformly at random from $[1, n-1]$ and computes $Q = dP$. The entity's *public key* is Q, while the entity's *private key* is d. Clearly, if the ECDLP is easy, then an adversary can deduce d from Q. Thus, the hardness of the ECDLP is crucial for the security of all elliptic curve schemes.

Although we have no proof that the ECDLP is indeed a hard problem, evidence for its hardness has been gathered over the years. First, the problem has been extensively studied by researchers for the last 16 years and no general-purpose subexponential-time algorithm has been discovered. This report will survey the work done on the ECDLP. Secondly, Shoup [60] has proved a lower bound of $\sqrt{n}$ for the discrete logarithm problem in a group of order n in the *generic group model*, i.e., when the group elements are random bit strings and one only has access to the group operation through a hypothetical oracle. While Shoup's result does not imply that the ECDLP is indeed hard, it does offer some hope that the DLP is hard in some groups.

On the negative side, it has been shown by Mosca and Ekert [49] that the ECDLP can be efficiently solved on a quantum computer. Of course, it is well known that the integer factorization and the ordinary discrete logarithm problems can also be efficiently solved on a quantum computer [59]. Thus, if large-scale quantum computers are ever built, then all the major families of public-key systems (DL, RSA, ECC) will be insecure. At present, this is not considered a serious concern since experts are still skeptical about whether quantum computers will ever be built. Hence our report does not take quantum computers into account.

2.2 Related Problems

Some elliptic curve cryptographic schemes, such as the elliptic curve Schnorr signature scheme (see Pointcheval and Stern [53]) can be proven secure under the assumption that the ECDLP is intractable (other assumptions not related to the elliptic curve are also necessary). Other protocols may require hardness of the elliptic curve Diffie-Hellman problem (ECDHP) or the elliptic curve decision Diffie-Hellman problem (ECDDHP).

Let E be an elliptic curve defined over $\mathbb{F}_q$, and let P be a point of order n on E. The ECDHP is the problem of finding abP given aP and bP. The ECDDHP is the problem, given aP, bP and cP, of deciding whether or not $c \equiv ab \pmod{n}$.

It is well known that the ECDDHP polynomial-time reduces to the ECDHP, and that the ECDHP polynomial-time reduces to the ECDLP. It would be useful to show that the ECDHP polynomial-time reduces to the ECDDHP and that the ECDLP polynomial-time reduces to the ECDHP because then we would have confidence that the ECDHP and ECDDHP are intractable based on our confidence that the ECDLP is intractable. Such results have not yet been proven. However, there is substantial evidence (see Maurer [42] and Maurer and Wolf [43]) that the ECDHP and ECDLP are indeed polynomial-time equivalent. In fact, Boneh and Lipton [8] have proven that if there is no subexponential-time algorithm (more precisely, one with running time $L_q[\frac{1}{2}]$) for

the ECDLP, then there also is no subexponential-time algorithm for the ECDHP. As a consequence, if we believe that the ECDLP has no subexponential-time algorithm, then we should also have confidence that the ECDHP is intractable.

The remainder of this report will not consider the ECDHP and ECDDHP, and will only consider algorithms for solving the ECDLP.

2.3 Overview of this Report

In Section 3, we provide basic terminology and results on elliptic curve over finite fields. We also define the special classes of finite fields and elliptic curves that have been proposed in various security standards. Section 4 and Section 5, respectively, survey the known general-purpose and special-purpose attacks on the ECDLP that have been discovered since the problem was first proposed 16 years ago in 1985. In Section 6, we discuss the security when special finite fields and special elliptic curves are used. Our conclusions are stated in Section 7.

3 Background on Elliptic Curves

We provide a brief introduction to finite fields and elliptic curves. The purpose is to remind the reader of the basic terminology and notation that will be used in the remainder of this report.

3.1 Finite Fields

DEFINITION. A *finite field* consists of a finite set of elements $\mathbb{F}$ together with two binary operations on $\mathbb{F}$, called addition and multiplication, that satisfy certain arithmetic properties. The *order* of a finite field is the number of elements in the field. There exists a finite field of order q if and only if q is a prime power. If q is a prime power, then there is essentially only one finite field of order q; this field is denoted by $\mathbb{F}_q$. If $q = p^m$ where p is a prime and m is a positive integer, then p is called the *characteristic* of $\mathbb{F}_q$, denoted $\mathrm{char}(\mathbb{F}_q)$, and m is called the *extension degree* of $\mathbb{F}_q$.

SPECIAL FIELDS. The following are some classes of finite fields that have been proposed for commercial use in elliptic curve cryptography due to their potential performance advantages.

1. *Prime fields*: These are finite fields of prime order. Prime fields have the advantage that their arithmetic can be efficiently implemented in software on machines which have a 32×32 bit multiply instruction.

2. *NIST prime fields*: These are prime fields $\mathbb{F}_p$ where the prime p is a Mersenne prime or a Mersenne-like prime, e.g., $p = 2^m - 2^k + 1$. In particular, the finite fields $\mathbb{F}_p$ for

$p = 2^{192} - 2^{64} - 1$, $p = 2^{224} - 2^{96} + 1$, $p = 2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$, $p = 2^{384} - 2^{128} - 2^{96} + 2^{32} - 1$, and $p = 2^{521} - 1$ have been standardized in NIST's FIPS 186-2 [50]. Such prime fields are advantageous over random prime fields because the modular reduction operation can be performed very efficiently (see Solinas [67]).

3. *Binary fields*: These are finite fields of order 2^m. Binary fields have the advantage that their arithmetic can be efficiently implemented in hardware. In particular, the binary fields $\mathbb{F}_{2^{163}}$, $\mathbb{F}_{2^{233}}$, $\mathbb{F}_{2^{283}}$, $\mathbb{F}_{2^{409}}$ and $\mathbb{F}_{2^{571}}$ have been standardized in NIST's FIPS 186-2 [50].

4. *Composite binary fields*: These are binary fields of order 2^m where m is a composite number. Because composite binary fields have non-trivial subfields, field arithmetic can be sped up by using lookup tables for performing subfield arithmetic; for example see [11].

5. *Optimal extension fields*: These are finite fields of order p^m where p is a 32-bit or 64-bit prime and m is a small integer. Optimal extension fields were introduced by Bailey and Paar [5] because the arithmetic in such fields is particularly efficient on 32-bit and 64-bit platforms.

3.2 Elliptic Curves

DEFINITION. Let $\mathbb{F}_q$ be a finite field of characteristic not equal to 3. If the characteristic of $\mathbb{F}_q$ is not equal to 2, then an elliptic curve over $\mathbb{F}_q$ is defined by an equation of the form

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

where $a, b \in \mathbb{F}_q$, and $4a^3 + 27b^2 \neq 0$. If the characteristic of $\mathbb{F}_q$ is equal to 2, then an elliptic curve over $\mathbb{F}_q$ is defined by an equation of the form

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

where $a, b \in \mathbb{F}_q$, $b \neq 0$, or by an equation of the form

$$y^2 + ay = x^3 + bx + c, \tag{3}$$

where $a, b, c \in \mathbb{F}_q$, $a \neq 0$. If the equation is (2) then the elliptic curve is said to have *non-zero j-invariant*, while if the equation is (3) then the elliptic curve is said to have *zero j-invariant*.

GROUP OF POINTS. The set $E(\mathbb{F}_q)$ consists of all points (x, y), $x \in \mathbb{F}_q$, $y \in \mathbb{F}_q$, which satisfy the defining equation, together with a special point O called the *point at infinity*. There is a rule, called the *chord-and-tangent rule*, for adding two points on an elliptic curve $E(\mathbb{F}_q)$ to give a third elliptic curve point. Together with this addition operation, the set of points $E(\mathbb{F}_q)$ forms a group with O serving as its identity. It is this group that is used in the construction of elliptic curve cryptographic schemes.

GROUP ORDER. Let E be an elliptic curve over a finite field $\mathbb{F}_q$. Hasse's theorem states that the number of points on an elliptic curve (including the point at infinity) is $\#E(\mathbb{F}_q) = q+1-t$ where $|t| \leq 2\sqrt{q}$; $\#E(\mathbb{F}_q)$ is called the *order* of E and t is called the *trace* of E. In other words, the order of an elliptic curve $E(\mathbb{F}_q)$ is roughly equal to the size q of the underlying field.

GROUP STRUCTURE. $E(\mathbb{F}_q)$ is an abelian group of rank 1 or 2; that is, $E(\mathbb{F}_q)$ is isomorphic to $\mathbb{Z}_{n_1} \times \mathbb{Z}_{n_2}$, where n_2 divides n_1, for unique positive integers n_1 and n_2. Here, $\mathbb{Z}_n$ denotes the cyclic group of order n. Moreover, n_2 divides $q-1$. If $n_2 = 1$, then $E(\mathbb{F}_q)$ is said to be *cyclic*. In this case $E(\mathbb{F}_q)$ is isomorphic to $\mathbb{Z}_{n_1}$, and there exists a point $P \in E(\mathbb{F}_q)$ such that $E(\mathbb{F}_q) = \{kP : 0 \leq k \leq n_1 - 1\}$; such a point is called a *generator* of $E(\mathbb{F}_q)$.

SPECIAL CURVES. The following are some classes of elliptic curves that have been proposed for commercial use in elliptic curve cryptography due to their potential performance advantages.

1. *Supersingular curves.* An elliptic curve E defined over a finite field $\mathbb{F}_q$ of characteristic p is said to be supersingular if $\#E(\mathbb{F}_q) = q+1-t$ where p divides t. Note that if $\mathbb{F}_q$ is a binary field, then the supersingular elliptic curves are precisely those with zero j-invariant (i.e., defined by equation (3)).

2. *Prime field anomalous curves.* An elliptic curve E defined over a prime field $\mathbb{F}_p$ is said to be prime field anomalous if $\#E(\mathbb{F}_p) = p$, i.e., the curve has trace 1.

3. *Koblitz curves.* A Koblitz curve E over $\mathbb{F}_{2^m}$ is an elliptic curve whose defining equation has coefficients in $\mathbb{F}_2$. There are two Koblitz curves: $y^2 + xy = x^3 + 1$ and $y^2 + xy = x^3 + x^2 + 1$. These elliptic curves were first proposed for cryptographic use by Koblitz [36]. They are advantageous over randomly selected curves over binary fields because the point multiplication operation in Koblitz curves involves no point doublings (see Solinas [66, 68]). Koblitz curves have been standardized in NIST's FIPS 186-2 [50].

4. *Elliptic curves with efficiently-computable endomorphisms.* Gallant, Lambert and Vanstone [25] showed how elliptic curves with efficiently-computable endomorphisms can be used to obtain a speedup of 50% for point multiplication. Examples of such elliptic curves include the Koblitz curves, elliptic curves $y^2 = x^3 + ax$ over prime fields $\mathbb{F}_p$ where $p \equiv 1 \pmod 4$, and elliptic curves $y^2 = x^3 + b$ over prime fields $\mathbb{F}_p$ where $p \equiv 1 \pmod 3$. One specific elliptic curve of this last type has been included in the WAP specification of the WTLS protocol [71].

5. *Elliptic curves with small class number.* Let E be an elliptic curve over a finite field $\mathbb{F}_q$, and let $\#E(\mathbb{F}_q) = q+1-t$. Such an elliptic curve is said to have small class number if its complex multiplication field $\mathbb{Q}(\sqrt{t^2-4q})$ has small class number. The complex multiplication (CM) method allows one to choose an elliptic curve group order before the equation of the curve is explicitly constructed. For elliptic curves over $\mathbb{F}_q$, the CM method is also called the Atkin-Morain method (see [48]); over $\mathbb{F}_{2^m}$, it is called the Lay-Zimmer method (see [38]). The CM method is very efficient for generating elliptic curves

of small class number; if there is no restriction on the class number then the CM method is extremely inefficient. Thus elliptic curves with small class number may arise in practice if the CM method has been used to generate them.

4 General-Purpose Attacks

Some algorithms for the ECDLP are tailored to perform better when the elliptic curve parameters are of a special form; these are called *special-purpose* attacks on the ECDLP. In contrast, the running times of the *general-purpose* attacks depend solely on the size of the elliptic curve parameters. This section describes the known general-purpose attacks on the ECDLP. Special-purpose attacks are considered in Section 5.

We recall the statement of the ECDLP. Let E be an elliptic curve defined over a finite field $\mathbb{F}_q$. Let $P \in E(\mathbb{F}_q)$ be a point of order n. Given E, $\mathbb{F}_q$, P, n and $Q \in \langle P \rangle$, the problem is to find the unique integer $l \in [0, n-1]$ such that $Q = lP$. Here, E, $\mathbb{F}_q$, P and n are called *elliptic curve parameters*.

4.1 Exhaustive Search

In exhaustive search, one simply computes successive multiples of P: P, $2P$, $3P$, $4P, \ldots$ until Q is obtained. This method can take up to n steps in the worst case. To circumvent this attack, one has to select elliptic curve parameters so that n is sufficiently large.

4.2 Pohlig-Hellman Attack

The Pohlig and Hellman attack [52], reduces the problem of recovering l to the problem of recovering l modulo each of the prime factors of n; the desired number l can then be recovered by using the Chinese Remainder Theorem.

The implications of this attack are the following. To construct the most difficult instance of the ECDLP, one must select an elliptic curve whose order is divisible by a large prime n. Preferably, this order should be a prime or almost a prime (i.e. a large prime n times a small integer h such as 2 or 4). For the remainder of this report, we shall assume that the order n of P is prime.

4.3 Pollard's Rho Algorithm

This algorithm, due to Pollard [54], is a randomized version of the baby-step giant-step algorithm. It has roughly the same expected running time ($\sqrt{\pi n}/2$ steps) as the baby-step giant-step algorithm, but is superior in that it requires a negligible amount of storage. Van Oorschot and

Wiener [51] showed how Pollard's rho algorithm can be parallelized so that when the algorithm is run in parallel on r processors, the expected running time of the algorithm is roughly $(\sqrt{\pi n})/(2r)$ steps. That is, using r processors results in an r-fold speed-up.

At present, parallelized Pollard's rho algorithm is the fastest general-purpose method for solving the ECDLP.

EXPERIMENTAL RESULTS. Certicom initiated an ECDLP challenge [10] in November 1997 in order to encourage and stimulate research on the ECDLP. Their challenges consist of instances of the ECDLP on a selection of elliptic curves. The challenge curves are divided into three categories listed below. In the following, ECCp-k denotes a randomly selected elliptic curve over a field $\mathbb{F}_p$, ECC2-k denotes a randomly selected elliptic curve over a field $\mathbb{F}_{2^m}$, and ECC2K-k denotes a Koblitz curve over $\mathbb{F}_{2^m}$; k is the bitlength of n. In all cases, the bitsize of the order of the underlying finite field is equal or slightly greater than k (so curves have either prime order or almost prime order).

1. Randomly generated curves over $\mathbb{F}_p$, where p is prime: ECCp-79, ECCp-89, ECCp-97, ECCp-109, ECCp-131, ECCp-163, ECCp-191, ECCp-239, and ECCp-359.
2. Randomly generated curves over $\mathbb{F}_{2^m}$, where m is prime: ECC2-79, ECC2-89, ECC2-97, ECC2-109, ECC2-131, ECC2-163, ECC2-191, ECC2-238, and ECC2-353.
3. Koblitz curves over $\mathbb{F}_{2^m}$, where m is prime: ECC2K-95, ECC2K-108, ECC2K-130, ECC2K-163, ECC2K-238, and ECC2K-358.

SOFTWARE IMPLEMENTATION. As of December 2001, the following challenges been solved:

- ECCp-79, ECCp-89, ECCp-97.
- ECC2-79, ECC2-89, ECC2-97.
- ECC2K-95, ECC2K-108.

Escott et al. [16] report on their 1998 implementation of the parallelized Pollard's rho algorithm which incorporates some improvements of Teske [70]. The hardest instance of the ECDLP they solved was the Certicom ECCp-97 challenge. For this task they utilized over 1200 machines from at least 16 countries, and found the answer in 53 days. The total number of steps executed was about 2×10^{14} elliptic curve additions which is close to the expected time ($(\sqrt{\pi n})/2 \approx 3.5 \times 10^{14}$, where $n \approx 2^{97}$). Escott et al. [16] conclude that the running time of Pollard's rho algorithm in practice fits well with the theoretical predictions. They estimate that the ECCp-109 challenge could be solved by a network of 50,000 Pentium Pro 200MHz machines in about 3 months. In April 2001, an Internet distributed effort was started to solve the ECCp-109 challenge [14]. As of December 2001, there were about 2000 machines from 60 teams around the world dedicated to solving this challenge, and about 13% of the solution space had been searched.

We can conclude from these efforts that 109-bit ECDLP instances are within the reach of software attacks by large organizations. However, the 163-bit challenges are well beyond the reach of software attacks for the forseeable future.

HARDWARE IMPLEMENTATION. Van Oorschot and Wiener [51] examined the feasibility of implementing parallelized Pollard's rho algorithm using special-purpose hardware. In their 1996 study, they estimated that if $n \approx 10^{36} \approx 2^{120}$, then a machine with $r = 330,000$ processors could be built for about US $10 million that could compute a single elliptic curve discrete logarithm in about 32 days. If the parameter n satisfies $n > 2^{160}$, then such hardware attacks are infeasible with today's technology.

SELECTING PARAMETERS FOR LONG-TERM SECURITY. Lenstra and Verheul [39] performed an extensive and careful study of the key sizes for symmetric-key encryption schemes, RSA, discrete logarithm systems, and ECC. Their study incorporates both software and hardware attacks, and takes into account the continual improvements in hardware, and hardware costs. Assuming that parallelized Pollard's rho algorithm remains the best algorithm for the ECDLP, they estimate that 163-bit ECC will provide the same level of security in the year 2021 as the Data Encryption Standard (DES) provided in the year 1982. (DES was considered very secure in 1982 for banking applications.) Similarly, 191-bit ECC will provide the same level of security in the year 2040 as DES provided in the year 1982.

4.4 Pollard's Lambda Algorithm

Like Pollard's rho method, the lambda method [54] can also be parallelized with a linear speedup. The parallelized lambda-method is slightly slower than the parallelized rho-method [51]. The lambda-method is, however, slightly faster in situations when the logarithm being sought is known to lie in a subinterval $[1,b]$ of $[1,n-1]$, where $b < 0.39n$ [51].

To circumvent this attack, private keys in elliptic curve systems should be selected uniformly at random from the interval $[1,n-1]$. That is, keys should not be selected in a special manner (e.g., by selecting small private keys) so that they are known a priori to lie in a subinterval of $[1,n-1]$.

4.5 Index-Calculus Attacks

As mentioned in Section 4.3, parallelized Pollard's rho algorithm is the best general-purpose algorithm known for solving the ECDLP and takes fully exponential time. It is natural to ask whether there exist subexponential-time "index-calculus" attacks such as the ones that are known for solving the ordinary discrete logarithm problem in the multiplicative group $\mathbb{F}_q^*$ of a finite field $\mathbb{F}_q$. By subexponential-time, we mean an algorithm whose running time is of the form

$$L_q[c,\alpha] = O\left(\exp\left((c+o(1))(\ln q)^{\alpha}(\ln\ln n)^{1-\alpha}\right)\right),$$

where c is a constant and $0 < \alpha < 1$. Recall that the number field sieve for the ordinary DLP in $\mathbb{F}_q^*$ has a running time of $L_q[1.923, \frac{1}{3}]$.

This question was asked by Miller in his paper introducing elliptic curve cryptography [47]. He observed that unlike in the case of $\mathbb{F}_q^*$, where there are natural candidates for the factor base Γ (prime numbers of small size in the case of prime fields or small degree irreducible polynomials in the case of binary fields), there appear to be no likely candidates in $E(\mathbb{F}_q)$. The most natural ones for elliptic curves over prime fields $\mathbb{F}_p$ seem to be points of small height in $E(\mathbb{Q})$, $\mathbb{Q}$ the field of rational numbers (the height of a point is related to the number of bits needed to represent the point). However, Miller points out that there are very few points of small height in $E(\mathbb{Q})$. Furthermore, even if such a set Γ exists, finding an efficient method for lifting a point in $E(\mathbb{F}_p)$ to a point in $E(\mathbb{Q})$ looks hopeless. Miller's argument against the possibility of index-calculus attacks has been elaborated on and explored in more detail by J. Silverman and Suzuki [62], who support his conclusions.

A very interesting line of attack on the ECDLP was recently proposed by J. Silverman [61]. His "xedni calculus" turns the index calculus method "on its head" (hence the name). Given a discrete logarithm problem on an elliptic curve over $\mathbb{F}_p$, he first lifts the points in question (actually, r different integer linear combinations of them, where $r \leq 9$) to points in the plane over $\mathbb{Q}$, and then he considers elliptic curves $E(\mathbb{Q})$ that pass through these r points. If $E(\mathbb{Q})$ can be chosen to have rank $< r$, i.e., so that there is an integer linear dependence relation among the r points—then the ECDLP is solved. In general, the probability of rank $< r$ is negligible. However, Silverman's idea is to impose a number of "Mestre conditions" modulo ℓ for small primes ℓ in order to increase this probability. (Each Mestre condition [46] forces $\#E(\mathbb{Z}_\ell)$ to be as small as possible.) Although the xedni calculus attack is clever and elegant, a careful analysis [33] showed that it is extremely impractical. One intriguing aspect of Silverman's algorithm is that it can be adapted (with no important changes) to solve both the discrete log problem in the multiplicative group of $\mathbb{F}_p$ and the integer factorization problem. Thus, if it had turned out to be efficient, it would have attacked all major public-key cryptosystems that are in practical use.

4.6 Multiple Logarithms

Suppose that we are given multiple instances of the ECDLP with respect to the same elliptic curve parameters. Such problems may arise in practice if all entities in a communications network are using the same domain parameters, but where each entity has its own public key.

One approach to tackling the multiple instances is to solve them iteratively using Pollard's rho method. R. Silverman and Stapleton [63] (see also Kuhn and Struik [37] for a formal analysis) observed that if a single instance of the ECDLP is solved using (parallelized) Pollard's rho method, then the work done in solving this instance can be used to speed up the solution of other instances of the ECDLP (for the same elliptic curve parameters). More precisely, if the first instance takes expected time t, then the second instance takes expected time $(\sqrt{2}-1)t \approx 0.41t$. Having solved these two instances, the third instance takes expected time $(\sqrt{3}-\sqrt{2})t \approx 0.32t$.

Having solved these three instances, the fourth instance takes expected time $(\sqrt{4}-\sqrt{3})t \approx 0.27t$. And so on. Thus subsequent instances of the ECDLP for a particular elliptic curve become progressively easier. Another way of looking at this is that solving k instances of the ECDLP (for the same curve E and base point P) takes only $\sqrt{k}$ as much work as it does to solve one instance of the ECDLP.

The other approach to tackling the multiple instances is to attack them simultaneously with the goal of solving *any one* instance. However, Kuhn and Struik [37] have proven that the best strategy for solving any instance is in fact to solve them iteratively.

Therefore, concerns that successive logarithms become easier can be addressed by ensuring that the elliptic parameters are chosen so that the first instance is infeasible to solve.

5 Special-Purpose Attacks

5.1 Weil Pairing and Tate Pairing Attacks

Menezes, Okamoto and Vanstone [44] and Frey and Rück [20] showed how, under mild assumptions, the ECDLP in an elliptic curve E defined over a finite field $\mathbb{F}_q$ can be reduced to the ordinary DLP in the multiplicative group of some extension field $\mathbb{F}_{q^k}$ for some $k \geq 1$, where the number field sieve algorithm applies. The first reduction is usually called the *MOV attack* or the *Weil pairing attack*, while the second reduction is usually called the *Frey-Rück attack* or the *Tate pairing attack*.

The reduction algorithms are only useful for solving the ECDLP if k is small—this is not the case for most elliptic curves, as shown by Balasubramanian and Koblitz [6]. To ensure that the reduction algorithm does not apply to a particular elliptic curve, one only needs to check that n, the order of the point P, does not divide $q^k - 1$ for all small k for which the DLP in $\mathbb{F}_{q^k}$ is tractable—in practice, when $n > 2^{160}$ then $1 \leq k \leq 20$ suffices.

However, it should be noted that the Weil and Tate pairing attacks are indeed useful for solving the ECDLP for special classes of elliptic curves. One class of such curves are the supersingular curves for which it is known that $k \leq 6$. The Weil and Tate pairing attacks yield a subexponential-time algorithm for the ECDLP in these curves. Another class of elliptic curves which succumbs to the Weil and Tate pairing attacks are curves of trace 2, i.e., elliptic curves E over $\mathbb{F}_q$ with $\#E(\mathbb{F}_q) = q - 1$. Hence these curves should not be used in practice.

We emphasize that the divisibility check rules out *all* elliptic curves which are susceptible to the Weil and Tate pairing attacks, including supersingular curve and trace 2 curves.

5.2 Prime Field Anomalous Curve Attack

Recall that an elliptic curve E over $\mathbb{F}_p$ is said to be *prime-field-anomalous* if $\#E(\mathbb{F}_p) = p$. Semaev [58], Smart [64], and Satoh and Araki [56] independently showed how to efficiently solve the ECDLP for these curves. The attack does *not* extend to any other classes of elliptic curves. Consequently, by verifying that the number of points on an elliptic curve is not equal to the cardinality of the underlying field, one can easily ensure that the Araki-Semaev-Smart-Satoh attack does not apply.

5.3 Speeding Up Pollard's Rho Algorithm for Koblitz Curves

Suppose that E is an elliptic curve defined over the finite field $\mathbb{F}_{2^e}$. Gallant, Lambert and Vanstone [24], and Wiener and Zuccherato [72] independently showed how Pollard's rho algorithm for computing elliptic curve logarithms in $E(\mathbb{F}_{2^{ed}})$ can be further sped up by a factor of $\sqrt{d}$—thus the expected running time of Pollard's rho method for these curves is $(\sqrt{\pi n/d})/2$ steps.

For example, if E is a Koblitz curve over $\mathbb{F}_{2^m}$, then Pollard's rho algorithm for computing elliptic curve logarithms in $E(\mathbb{F}_{2^m})$ can be sped up by a factor of $\sqrt{m}$. This speedup is not a concern in practice since the factor $\sqrt{m}$ is relatively small (e.g., for $m = 163$, $\sqrt{m} \approx 13$). Nonetheless, this speedup should be considered when doing a security analysis of elliptic curves whose coefficients lie in a small subfield.

5.4 Weil Descent

In this section, we say that an elliptic curve E defined over the binary field $\mathbb{F}_{2^N}$ is *cryptographically interesting* if (i) $\#E(\mathbb{F}_{2^N})$ is almost prime—that is, $\#E(\mathbb{F}_{2^N}) = rd$ where r is prime and $d \in \{2, 4\}$ (in order to avoid the Pohlig-Hellman and Pollard's rho attacks); and (ii) r does not divide $2^{Nj} - 1$ for each $j \in [1, J]$, where J is large enough so that it is computationally infeasible to find discrete logarithms in $\mathbb{F}_{2^{NJ}}$ (in order to avoid the Weil pairing and Tate pairing attacks).

Frey [18, 19] first proposed using Weil descent as a means to reduce the ECDLP in elliptic curves over binary fields $\mathbb{F}_{2^N}$ to the discrete logarithm problem in an abelian variety over a proper subfield $\mathbb{F}_{2^l}$ of $\mathbb{F}_{2^N}$. Frey's method, which we refer to as the *Weil descent attack methodology*, was further elaborated by Galbraith and Smart [23]. In 2000, Gaudry, Hess and Smart (GHS) [27] showed how Frey's methodology could be used (in most cases) to reduce any instance of the ECDLP to an instance of the discrete logarithm problem in the Jacobian of a hyperelliptic curve over $\mathbb{F}_{2^l}$. Since subexponential-time algorithms for the hyperelliptic curve discrete logarithm problem (HCDLP) are known (see [1], [26], and [15]), this could have important implications to the security of elliptic curve cryptographic schemes.

The GHS attack was analyzed by Menezes and Qu [45]. It was proven to fail for *all* cryptographically interesting elliptic curves over $\mathbb{F}_{2^N}$, where $N \in [160, 600]$ is prime. Namely, the

hyperelliptic curves C produced either have genus too small (whence $J_C(\mathbb{F}_2)$ is too small to yield any non-trivial information about the ECDLP in $E(\mathbb{F}_{2^N})$), or have genus too large ($g \geq 2^{16}-1$, whence the HCDLP in $J_C(\mathbb{F}_2)$ is infeasible using known methods for solving the HCDLP).

However, the GHS attack *is* effective for solving the ECDLP on some elliptic curves over composite binary fields. Jacobson, Menezes and Stein [34] implemented the GHS attack and demonstrated that the ECDLP on a certain class of elliptic curves over $\mathbb{F}_{2^{155}}$ is feasible (see also Smart [65] for a partial analysis). The ECDLP on elliptic curves in this class can be solved in about one month using a network of 1,000 1 GHz Pentium III workstations (see [34]). This is the same order of magnitude as the work required to perform exhaustive search on the DES key space (estimated time is 110,000 days on a single 450 MHz Pentium PC [29]), and less than the estimated time of 200,000 days on a single 450 MHz Pentium PC spent on the Certicom ECC2-108K ECDLP challenge [29].

The GHS attack is effective on only about 2^{32} out of the 2^{156} isomorphism classes of elliptic curves over $\mathbb{F}_{2^{155}}$. However, Galbraith, Hess and Smart [22] (see also [21]) very recently presented an algorithm with expected average running time of $O(q^{n/4+\varepsilon})$ for explicitly computing an isogeny between two isogenous elliptic curve over $\mathbb{F}_{q^n}$. (Two elliptic curves $E_1/\mathbb{F}_{q^n}$ and $E_2/\mathbb{F}_{q^n}$ are said to be *isogenous* over $\mathbb{F}_{q^n}$ if $\#E_1(\mathbb{F}_{q^n}) = \#E_2(\mathbb{F}_{q^n})$.) They observed that this algorithm can be used to extend the effectiveness of the GHS attack as follows. Given an ECDLP instance on some cryptographically interesting elliptic curve $E_1/\mathbb{F}_{2^N}$, one can check if E_1 is isogenous to some elliptic curve $E_2/\mathbb{F}_{2^N}$ which yields an easier HCDLP than E_1, and then use an isogeny $\phi : E_1 \to E_2$ to map the ECDLP instance to an ECDLP instance in $E_2(\mathbb{F}_{2^N})$. For example, in the case $N = 155$, we can expect that roughly 2^{104} out of 2^{156} elliptic curves over $\mathbb{F}_{2^{155}}$ are isogenous to one of the $\approx 2^{32}$ elliptic curves over $\mathbb{F}_{2^{155}}$ originally believed to be susceptible to the GHS attack. Thus, the GHS attack is now known to be effective on 2^{104} out of the 2^{156} elliptic curves over $\mathbb{F}_{2^{155}}$.

The GHS attack on elliptic curves over composite binary fields was fully analyzed by Maurer, Menezes and Teske [41]. They identify all binary fields $\mathbb{F}_{2^N}$ where $N \in [100, 600]$ is a composite integer, for which the GHS attack succeeds for *some* elliptic curves over $\mathbb{F}_{2^N}$. Such fields include $\mathbb{F}_{2^{155}}$, $\mathbb{F}_{2^{161}}$, $\mathbb{F}_{2^{180}}$, $\mathbb{F}_{2^{186}}$, $\mathbb{F}_{2^{217}}$, $\mathbb{F}_{2^{248}}$ and $\mathbb{F}_{2^{300}}$. For each such field, they list the number of elliptic curves that succumb to the GHS attack. There are also some fields, such as $\mathbb{F}_{2^{185}}$, $\mathbb{F}_{2^{215}}$ and $\mathbb{F}_{2^{265}}$ for which the GHS attack fails for *all* elliptic curves over that field.

For most composite binary fields $\mathbb{F}_{2^N}$, the proportion of elliptic curves over $\mathbb{F}_{2^N}$ which succumb to the GHS attack is very small. For example, the proportion of elliptic curves over $\mathbb{F}_{2^{155}}$ which succumb to the GHS attack is only $\frac{1}{2^{52}}$. Thus, if one selects an elliptic curve at random, then there is a very high probability that the elliptic curve will be resistant to the GHS attack. However, failure of the GHS attack does not imply failure of the Weil descent methodology—there may be other useful curves which lie on the Weil restriction that were not constructed by the GHS method. Thus, to take into account potential future developments in the Weil descent methodology, it is prudent to altogether avoid using elliptic curves over composite binary fields.

Arita [4] showed that some elliptic curves over finite fields $\mathbb{F}_{3^m}$ may also be susceptible to the

Weil descent attack. However, since elliptic curves over $\mathbb{F}_{3^m}$ have never been proposed for commercial applications, we will not consider them any further in this report.

Diem [12, 13] has shown that the GHS attack can be extended to elliptic curves over $\mathbb{F}_{p^m}$ where $p \geq 5$ is prime. He concludes that his particular variant of the GHS attack will always fail when m is prime and $m \geq 11$—that is, the discrete logarithm problem in the resulting higher-genus curves is intractable. However, he provides some evidence that the attack *may* succeed for *some* elliptic curves when $m = 5$ or $m = 7$. Further research and experimentation is necessary before one can conclude this with certainty.

We conclude this section by emphasizing that the GHS attack is *not* applicable to elliptic curves over binary fields $\mathbb{F}_{2^m}$ where m is prime, or to elliptic curves over prime fields $\mathbb{F}_p$.

6 Special Parameters

In this section, we discuss the hardness of the ECDLP when special finite fields and special elliptic curves are used. Such special finite fields and elliptic curves are typically used because they allow for some performance enhancements. For an indication of the improved performances that are achievable by using the NIST prime fields or Koblitz curves, see the implementation reports [9] and [28].

6.1 NIST Prime Fields

Recall that the NIST prime fields are fields $\mathbb{F}_p$ where the prime p is a Mersenne prime or a Mersenne-like prime, e.g., $p = 2^m - 2^k + 1$. There are no known attacks on the ECDLP that exploit the special form of the primes in the NIST prime fields. Thus, the NIST primes should be considered to be a safe alternative to randomly generated primes.

6.2 Composite Binary Fields

Recall that a composite binary field is a binary field $\mathbb{F}_{2^N}$ where N is composite. As discussed in Section 5.4, for most composite binary fields $\mathbb{F}_{2^N}$, the proportion of elliptic curves over $\mathbb{F}_{2^N}$ which succumb to the GHS attack is very small. Indeed there are some composite binary fields for which the GHS attack fails for *all* elliptic curves over that field.

Of special interest are elliptic curves over $\mathbb{F}_{2^{155}}$ because a specific elliptic curve over $\mathbb{F}_{2^{155}}$ has been included in an internet standard for key agreement [32]. Since the fraction of elliptic curves over $\mathbb{F}_{2^{155}}$ which succumb to the GHS attack is only $\frac{1}{2^{52}}$, the GHS attack can be safely avoided by using a randomly generated elliptic curve over $\mathbb{F}_{2^{155}}$.

However, as stated in Section 5.4, failure of the GHS attack does not imply failure of the Weil

descent methodology—there may be other useful curves which lie on the Weil restriction that were not constructed by the GHS method. Thus, to take into account potential future developments in the Weil descent methodology, it is prudent to altogether avoid using elliptic curves over composite binary fields. (Note: This is a conservative recommendation.)

6.3 Optimal Extension Fields

Recall that optimal extension fields are finite fields $\mathbb{F}_{p^m}$ where p is a 32-bit or 64-bit prime and m is a small integer. Some examples of optimal extension fields that have been implemented have a 32-bit prime p and $m = 5$ or $m = 7$ (see [5]). As mentioned in Section 5.4, The GHS Weil descent *may* succeed for some elliptic curves over optimal extension fields with $m = 5$ or $m = 7$; however more research and experimentation is necessary before this can be concluded with certainty. To take into account potential future developments in the Weil descent methodology, it is prudent to altogether avoid using optimal extension fields. (Note: This is a conservative recommendation.)

6.4 Koblitz Curves

The only special-purpose attack on the ECDLP for Koblitz curves over $\mathbb{F}_{2^m}$ (where m is prime) is the factor $\sqrt{m}$-speedup in Pollard's rho algorithm that was noted in Section 5.3. Since this attack only reduces the time required to compute elliptic curve logarithms by a small factor, it is not a practical concern. Thus, Koblitz curves should be considered to be a safe alternative to randomly generated elliptic curves over $\mathbb{F}_{2^m}$. Indeed, Koblitz curves have been adopted by the U.S. government in the FIPS 186-2 standard [50] for the elliptic curve digital signature algorithm (ECDSA).

6.5 Elliptic Curves with Efficiently-Computable Endomorphisms

The only special-purpose attack on the ECDLP for elliptic curves with efficiently-computable endomorphisms are the improvements to Pollard's rho algorithm that are similar to the ones noted in Section 5.3 for Koblitz curves (see [24] and [72]). Since this attack only reduces the time required to compute elliptic curve logarithms by a small factor, it is not a practical concern. Thus, elliptic curves with efficiently-computable endomorphisms should be considered to be a safe alternative to randomly generated elliptic curves.

6.6 Elliptic Curves with Small Class Number

Recall that elliptic curves with small class number are typically produced by the Lay-Zimmer and Atkin-Morain methods. There has been some concern expressed by experts that the small

class number might lead to attacks on the ECDLP for these elliptic curves. However, no such attacks have ever been proposed.

In any case, in the last few years there have been some dramatic improvements in Schoof's original algorithm [57] for counting the points on a randomly generated elliptic curve over finite fields (for example, see Fouquet, Gaudry and Harley [17], Lercier and Morain [40], and Satoh [55]). With these algorithms, it is relatively easy to count the number of points on a randomly selected elliptic curve. Therefore, there is no longer any compelling reason to use the Lay-Zimmer or the Atkin-Morain methods in practice—in this way, elliptic curves with small class number will not arise in practice.

7 Conclusions

Table 1 summarizes the known attacks on the ECDLP and countermeasures for ensuring that a given elliptic curve is immune to these attacks. Recall that elliptic curve parameters consist of an elliptic curve E defined over a finite field $\mathbb{F}_q$, and a point $P \in E(\mathbb{F}_q)$ of order n.

Attack	Countermeasure
Pohlig-Hellman (Section 4.2)	Select n to be prime.
Pollard-rho (Section 4.3)	Select n so that $\sqrt{n}$ represents an infeasible amount of computation. At a minimum, n should be at least 2^{160}.
Multiple logarithms (Section 4.6)	Select n so that $\sqrt{n}$ represents an infeasible amount of computation. At a minimum, n should be at least 2^{160}.
Weil pairing and Tate pairing attacks (Section 5.1)	Check that n does not divide $q^k - 1$ for all $1 \leq k \leq 20$. (This rules out supersingular and trace 2 elliptic curves.)
Prime field anomalous curve attack (Section 5.2)	Check that $n \neq q$.
Weil descent attack (Section 5.4)	Do not use elliptic curves over composite binary fields. Do not use elliptic curves over $\mathbb{F}_{p^m}$ where p is odd and $m = 5$ or $m = 7$. (Note: These are quite conservative recommendations.)

Table 1: Summary of attacks on the ECDLP and countermeasures.

If an elliptic curve is selected that meets all the requirements in Table 1, then the ECDLP is intractable against all known attacks.

References

[1] L. Adleman, J. DeMarrais and M. Huang, "A subexponential algorithm for discrete logarithms over the rational subgroup of the jacobians of large genus hyperelliptic curves over finite fields", *Algorithmic Number Theory*, Lecture Notes in Computer Science, **877** (1994), Springer-Verlag, 28-40.

[2] ANSI X9.62, *Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)*, 1999.

[3] ANSI X9.63, *Public Key Cryptography for the Financial Services Industry: Elliptic Curve Key Agreement and Key Transport Protocols*, ballot version, May 2001.

[4] S. Arita, "Weil descent of elliptic curves over finite fields of characteristic three", *Advances in Cryptology–Asiacrypt 2000*, Lecture Notes in Computer Science, **1976** (2000), Springer-Verlag, 248-259.

[5] D. Bailey and C. Paar, "Efficient arithmetic in finite field extensions with application in elliptic curve cryptography", *Journal of Cryptology*, **14** (2001), 153-176.

[6] R. Balasubramanian and N. Koblitz, "The improbability that an elliptic curve has subexponential discrete log problem under the Menezes–Okamoto–Vanstone algorithm", *Journal of Cryptology*, **11** (1998), 141-145.

[7] L. Bassham, D. Johnson and T. Polk, *Representation of Elliptic Curve Digital Signature Algorithm (ECDSA) Keys and Signatures in Internet X.509 Public Key Infrastructure Certificates*, Internet Draft, June 1999. Available at http://www.ietf.org

[8] D. Boneh and R. Lipton, "Algorithms for black-box fields and their application to cryptography", *Advances in Cryptology–Crypto '96*, Lecture Notes in Computer Science, **1109** (1996), Springer-Verlag, 283-297.

[9] M. Brown, D. Hankerson, J. Hernandez and A. Menezes, "Software implementation of the NIST elliptic curves over prime fields", *Topics in Cryptology–CT-RSA 2001*, Lecture Notes in Computer Science, **2020** (2001), Springer-Verlag, 250-265.

[10] Certicom ECC Challenge, November 1997, http://www.certicom.com

[11] E. De Win, A. Bosselaers, S. Vandenberghe, P. De Gersem and J. Vandewalle, "A fast software implementation for arithmetic operations in $GF(2^n)$", *Advances in Cryptology–Asiacrypt '96*, Lecture Notes in Computer Science, **1163** (1996), Springer-Verlag, 65-76.

[12] C. Diem, *A Study on Theoretical and Practical Aspects of Weil-Restrictions of Varieties*, Ph.D. thesis, University of Essen, 2001.

[13] C. Diem, "The GHS-attack in odd characteristic", preprint, 2001. Available from http://www.exp-math.uni-essen.de/~diem/english.html

[14] ECCp-109 Challenge, http://www.nd.edu/~cmonico/eccp109

[15] A. Enge and P. Gaudry, "A general framework for subexponential discrete logarithm algorithms", *Acta Arithmetica*, to appear.

[16] A. Escott, J. Sager, A. Selkirk and D. Tsapakidis, "Attacking elliptic curve cryptosystems using the parallel Pollard rho method", *CryptoBytes – The Technical Newsletter of RSA Laboratories*, volume 4, number 2, Winter 1999, 15-19. Also available at http://www.rsasecurity.com

[17] M. Fouquet, P. Gaudry and R. Harley, "An extension of Satoh's algorithm and its implementation", *Journal of the Ramanujan Mathematical Society*, **15** (2000), 281-318.

[18] G. Frey, "How to disguise an elliptic curve (Weil descent)", Talk at ECC '98, Waterloo, 1998. Slides available from http://www.cacr.math.uwaterloo.ca/conferences/1998/ecc98/slides.html

[19] G. Frey, "Applications of arithmetical geometry to cryptographic constructions", *Proceedings of the Fifth International Conference on Finite Fields and Applications*, Springer-Verlag, 2001, 128-161.

[20] G. Frey and H. Rück, "A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves", *Mathematics of Computation*, **62** (1994), 865-874.

[21] S. Galbraith, "Constructing isogenies between elliptic curves over finite fields", *LMS Journal of Computation and Mathematics*, **2** (1999), 118-138.

[22] S. Galbraith, F. Hess and N. Smart, "Extending the GHS Weil descent attack", preprint, 2001.

[23] S. Galbraith and N. Smart, "A cryptographic application of Weil descent", *Codes and Cryptography*, Lecture Notes in Computer Science, **1746** (1999), Springer-Verlag, 191-200.

[24] R. Gallant, R. Lambert and S. Vanstone, "Improving the parallelized Pollard lambda search on anomalous binary curves", *Mathematics of Computation*, **69** (2000), 1699-1705.

[25] R. Gallant, R. Lambert and S. Vanstone, "Faster point multiplication on elliptic curves with efficient endomorphisms", *Advances in Cryptology–Crypto 2001*, Lecture Notes in Computer Science, **2139** (2001), Springer-Verlag, 190-200.

[26] P. Gaudry, "An algorithm for solving the discrete log problem on hyperelliptic curves", *Advances in Cryptology–Eurocrypt 2000*, Lecture Notes in Computer Science, **1807** (2000), Springer-Verlag, 19-34.

[27] P. Gaudry, F. Hess and N. Smart, "Constructive and destructive facets of Weil descent on elliptic curves", *Journal of Cryptology*, to appear.

[28] D. Hankerson, J. Hernandez and A. Menezes, "Software implementation of elliptic curve cryptography over binary fields", *Proceedings of CHES 2000*, Lecture Notes in Computer Science, **1965** (2000), Springer-Verlag, 1-24.

[29] R. Harley, Fact sheet for solution of ECC2-108K ECDLP challenge, http://cristal.inria.fr/~harley/ecdl7/factsheet.html

[30] IEEE 1363-2000, *Standard Specifications for Public-Key Cryptography*, 2000. http://grouper.ieee.org/groups/1363/index.html

[31] ISO/IEC 15946-3, *Information Technology–Security Techniques–Cryptographic Techniques Based on Elliptic Curves, Part 3*, Final Draft International Standard (FDIS), February 2001.

[32] Internet Engineering Task Force, *The OAKLEY Key Determination Protocol*, IETF RFC 2412, November 1998.

[33] M. Jacobson, N. Koblitz, J. Silverman, A. Stein and E. Teske, "Analysis of the xedni calculus attack", *Designs, Codes and Cryptography*, **20** (2000), 41-64.

[34] M. Jacobson, A. Menezes and A. Stein, "Solving elliptic curve discrete logarithm problems using Weil descent", *Journal of the Ramanujan Mathematical Society*, **16** (2001), 231-260.

[35] N. Koblitz, "Elliptic curve cryptosystems", *Mathematics of Computation*, **48** (1987), 203-209.

[36] N. Koblitz, "CM-curves with good cryptographic properties", *Advances in Cryptology–Crypto '91*, Lecture Notes in Computer Science, **576** (1992), Springer-Verlag, 279-287.

[37] F. Kuhn and R. Struik, "Random walks revisited: Extensions of Pollard's rho algorithm for computing multiple discrete logarithms", *Selected Areas in Cryptography–Proceedings of SAC 2001*, Lecture Notes in Computer Science, to appear.

[38] G. Lay and H. Zimmer, "Constructing elliptic curves with given group order over large finite fields", *Algorithmic Number Theory*, Lecture Notes in Computer Science, **877** (1994), Springer-Verlag, 250-263.

[39] A. Lenstra and E. Verheul, "Selecting cryptographic key sizes", *Public Key Cryptography–Proceedings of PKC 2000*, Lecture Notes in Computer Science, **1751** (2000), Springer-Verlag, 446-465. Full version to appear in *Journal of Cryptology*.

[40] R. Lercier and F. Morain, "Counting the number of points on elliptic curves over finite fields: strategies and performances", *Advances in Cryptology–Eurocrypt '95*, Lecture Notes in Computer Science, **921** (1995), Springer-Verlag, 79-94.

[41] M. Maurer, A. Menezes and E. Teske, "Analysis of the GHS Weil descent attack on the ECDLP over characteristic two finite fields of composite degree", *Advances in Cryptology–Indocrypt 2001*, to appear.

[42] U. Maurer, "Towards the equivalence of breaking the Diffie-Hellman protocol and computing discrete logarithms", *Advances in Cryptology–Crypto '94*, Lecture Notes in Computer Science, **839** (1994), Springer-Verlag, 271-281.

[43] U. Maurer and S. Wolf, "The Diffie-Hellman protocol", *Designs, Codes and Cryptography*, **19** (2000), 147-171.

[44] A. Menezes, T. Okamoto and S. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field", *IEEE Transactions on Information Theory*, **39** (1993), 1639-1646.

[45] A. Menezes and M. Qu, "Analysis of the Weil descent attack of Gaudry, Hess and Smart", *Topics in Cryptology–CT-RSA 2001*, Lecture Notes in Computer Science, **2020** (2001), Springer-Verlag, 308-318.

[46] J. Mestre, "Formules explicites et minoration de conducteurs de variétés algébriques," *Compositio Math.*, **58** (1986), 209-232.

[47] V. Miller, "Uses of elliptic curves in cryptography", *Advances in Cryptology–Crypto '85*, Lecture Notes in Computer Science, **218** (1986), Springer-Verlag, 417-426.

[48] F. Morain, "Building cyclic elliptic curves modulo large primes", *Advances in Cryptology–Eurocrypt '91*, Lecture Notes in Computer Science, **547** (1991), Springer-Verlag, 328-336.

[49] M. Mosca and A. Ekert, "The hidden subgroup problem and eigenvalue estimation on a quantum computer", *Proceedings of the 1st NASA International Conference on Quantum Computing and Quantum Communication*, Lecture Notes in Computer Science, **1509** (1999), Springer-Verlag.

[50] National Institute of Standards and Technology, *Digital Signature Standard*, FIPS Publication 186-2, 2000.

[51] P. van Oorschot and M. Wiener, "Parallel collision search with cryptanalytic applications", *Journal of Cryptology*, **12** (1999), 1-28.

[52] S. Pohlig and M. Hellman, "An improved algorithm for computing logarithms over $GF(p)$ and its cryptographic significance", *IEEE Transactions on Information Theory*, **24** (1978), 106-110.

[53] D. Pointcheval and J. Stern, "Security proofs for signature schemes", *Advances in Cryptology–Eurocrypt '96*, Lecture Notes in Computer Science, **1070** (1993), Springer-Verlag, 387-398.

[54] J. Pollard, "Monte Carlo methods for index computation mod p", *Mathematics of Computation*, **32** (1978), 918-924.

[55] T. Satoh, "The canonical lift of an ordinary elliptic curve over a prime field and its point counting", *Journal of the Ramanujan Mathematical Society*, **15** (2000), 247-270.

[56] T. Satoh and K. Araki, "Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves", *Commentarii Mathematici Universitatis Sancti Pauli*, **47** (1998), 81-92.

[57] R. Schoof, "Elliptic curves over finite fields and the computation of square roots mod p", *Mathematics of Computation*, **44** (1985), 483-494.

[58] I. Semaev, "Evaluation of discrete logarithms in a group of p-torsion points of an elliptic curve in characteristic p", *Mathematics of Computation*, **67** (1998), 353-356.

[59] Peter Shor, "Algorithms for quantum computation: Discrete logarithms and factoring", *Proceedings of the 35nd Annual Symposium on Foundations of Computer Science*, IEEE Computer Society Press (1994), 124-134.

[60] V. Shoup, "Lower bounds for discrete logarithms and related problems", *Advances in Cryptology–Eurocrypt '97*, Lecture Notes in Computer Science, **1233** (1997), Springer-Verlag, 256-266.

[61] J. Silverman, "The xedni calculus and the elliptic curve discrete logarithm problem", *Designs, Codes and Cryptography*, **20** (2000), 5-40.

[62] J. Silverman and J. Suzuki, "Elliptic curve discrete logarithms and the index calculus", *Advances in Cryptology–Asiacrypt '98*, Lecture Notes in Computer Science, **1514** (1999), Springer-Verlag, 110-125.

[63] R. Silverman and J. Stapleton, Contribution to ANSI X9F1 working group, 1997.

[64] N. Smart, "The discrete logarithm problem on elliptic curves of trace one", *Journal of Cryptology*, **12** (1999), 193-196.

[65] N. Smart, "How secure are elliptic curves over composite extension fields?", *Advances in Cryptology–Eurocrypt 2001*, Lecture Notes in Computer Science, **2045** (2001), Springer-Verlag, 30-39.

[66] J. Solinas, "An improved algorithm for arithmetic on a family of elliptic curves", *Advances in Cryptology–Crypto '97*, Lecture Notes in Computer Science, **1294** (1997), Springer-Verlag, 357-371.

[67] J. Solinas, "Generalized Mersenne numbers", Technical report CORR 99-39, Dept. of C&O, University of Waterloo, 1999. Available from http://www.cacr.math.uwaterloo.ca

[68] J. Solinas, "Efficient arithmetic on Koblitz curves", *Designs, Codes and Cryptography*, **19** (2000), 195-249.

[69] Standards for Efficient Cryptography Group, *SEC 1: Elliptic Curve Cryptography*, version 1.0, 2000. Available at http://www.secg.org

[70] E. Teske, "Speeding up Pollard's rho method for computing discrete logarithms", *Algorithmic Number Theory*, Lecture Notes in Computer Science, **1423** (1998), Springer-Verlag, 541-554.

[71] WAP WTLS, *Wireless Application Protocol Wireless Transport Layer Security Specification*, Wireless Application Protocol Forum, February 1999. Drafts available at http://www.wapforum.org

[72] M. Wiener and R. Zuccherato, "Faster attacks on elliptic curve cryptosystems", *Selected Areas in Cryptography*, Lecture Notes in Computer Science, **1556** (1999), Springer-Verlag, 190-200.