Towards the Equivalence of Breaking the Diffie-Hellman Protocol and Computing Discrete Logarithms[*]

Ueli M. Maurer

Institute for Theoretical Computer Science
ETH Zürich
CH-8092 Zürich, Switzerland
Email address: maurer@inf.ethz.ch

Abstract. Let G be an arbitrary cyclic group with generator g and order $|G|$ with known factorization. G could be the subgroup generated by g within a larger group H. Based on an assumption about the existence of smooth numbers in short intervals, we prove that breaking the Diffie-Hellman protocol for G and base g is equivalent to computing discrete logarithms in G to the base g when a certain side information string S of length $2 \log |G|$ is given, where S depends only on $|G|$ but not on the definition of G and appears to be of no help for computing discrete logarithms in G. If every prime factor p of $|G|$ is such that one of a list of expressions in p, including $p-1$ and $p+1$, is smooth for an appropriate smoothness bound, then S can efficiently be constructed and therefore breaking the Diffie-Hellman protocol is equivalent to computing discrete logarithms.

1. Introduction

Two challenging open problems in cryptography are to prove or disprove that breaking the Diffie-Hellman protocol [5] is computationally equivalent to computing discrete logarithms in the underlying group and that breaking the RSA system [17] is computationally equivalent to factoring the modulus. In this paper we take a significant step towards the solution of the first of these problems.

Let H be a finite group (written multiplicatively), and for $g \in H$, let $G = \langle g \rangle$ be the cyclic subgroup generated by g. The discrete logarithm problem for the group H (or G) can be stated as follows: Given g and $a \in G$, find the unique integer x in the interval $[0, |G|-1]$ such that $g^x = a$, where x is called the discrete logarithm of a to the base g. The discrete logarithm problem is sometimes also defined as the generally easier problem of finding any x satisfying $g^x = a$, but if $|G|$ is known then the two problems are equivalent.

The Diffie-Hellman protocol allows two parties Alice and Bob connected by an authenticated but otherwise insecure channel (for instance an insecure telephone

[*] Appeared in *Advances in Cryptology – CRYPTO '94*, Y. Desmedt (Ed.), Lecture Notes in Computer Science, Berlin: Springer-Verlag, vol. 839, pp. 271-281, 1994.

line where Alice and Bob authenticate each other by speaker recognition) to generate a mutual secret key which is computationally infeasible to determine for a passive eavesdropper overhearing the entire conversation between Alice and Bob.

The protocol works as follows. Let $G = \langle g \rangle$ be a cyclic group generated by g for which the discrete logarithm problem is computationally infeasible. (It should be pointed out that it is unknown whether such a group exists.) Specific groups that have been proposed for application in this protocol are the multiplicative groups of large finite fields (prime fields [5] or extension fields), the multiplicative group of residues modulo a composite number [10, 11], elliptic curves over finite fields, the Jacobian of a hyperelliptic curve over a finite field and the class group of imaginary quadratic fields [2].

In order to generate a mutual secret key, Alice and Bob secretly choose integers x_A and x_B, respectively, at random from the interval $[0, |G| - 1]$. Then they compute secretly $y_A = g^{x_A}$ and $y_B = g^{x_B}$, respectively, and exchange these group elements over the insecure public channel. Finally, they compute $z_{AB} = y_B^{x_A} = g^{x_A x_B}$ and $z_{BA} = y_A^{x_B} = g^{x_B x_A}$, respectively. Note that $z_{AB} = z_{BA}$, and hence this quantity can be used as a secret key shared by Alice and Bob. More precisely, they need to apply a function mapping elements of G to the key space of a cryptosystem.

In contrast to digital signature schemes based on the discrete logarithm problem (e.g., [6],[19]), it is not required for the Diffie-Hellman protocol that the order of the group be known. In this case, x_A and x_B are chosen from a sufficiently large interval. In fact, it has been pointed out (e.g., see [11]) that using groups with unknown order may be advantageous, and the non-interactive public-key scheme of [10] relies crucially on the fact that the group order is unknown.

2. Computing discrete logarithms and breaking the Diffie-Hellman protocol

An eavesdropper knowing y_A and y_B can in principle compute z_{AB} by computing the discrete logarithm x_A of y_A to the base g and then computing $z_{AB} = y_B^{x_A}$. It is unknown in general whether there exists a faster method for computing z_{AB} from y_A and y_B. This paper investigates the relation between the two problems. More precisely, we investigate in which cases an efficient subroutine breaking the Diffie-Hellman protocol for the group G and base g could be used for computing discrete logarithms to the base g in G.

Definition 1. A *Diffie-Hellman oracle* (DH-oracle for short) for a group G with generator g takes as inputs two elements $a, b \in G$ (where $a = g^x$ and $b = g^y$) and returns the element g^{xy}.

A DH-oracle hence allows to multiply two logarithms without knowing them explicitly but also without receiving the result explicitly. For instance, when given g^x but not x one can use the oracle to compute $g^{(x^2)}$, and more generally $g^{P(x)}$ for any polynomial P with integer coefficients. Multiplications and additions in the exponent are performed by using the oracle and the normal group multiplication, respectively. Multiplication of the exponent with -1 can be achieved by an inversion operation in the group. When $|G|$ is known and $\gcd(x, |G|) = 1$ one can compute g^z from g^x such that $z = x^{-1} \pmod{|G|}$ by computing $g^z = g^{(x^{|G|-1})}$ using $O(\log |G|)$ calls to the DH-oracle. Hence the DH-oracle allows one to compute $g^{f(x)}$ for any rational function $f(x)$ with integer coefficients. More generally, such a DH-oracle can be used to perform any algorithm on implicitly given (but hidden) logarithms, provided that the algorithm uses only addition, subtraction, multiplication and makes decisions only based on testing equality of intermediate results.

For instance, one can compute g^z from g^x where $z^2 = x \pmod{|G|}$ by using the algorithm of [14]. (A more efficient but unpublished algorithm is due to Massey [9].) For the case $|G| = p$ with p prime and $p \equiv 3 \pmod 4$ one can compute $g^z = g^{(x^{(p+1)/4})}$. Our proof techniques will exploit these facts.

When proving a reduction of a problem A to another problem B it is important to state precisely what type of instances of B are generated by the reduction process. If, in the process of solving problem A, the instances of problem B that need to be solved are very special, then the reduction from A to B is not satisfactory because it is conceivable that these special instances are easy to solve even if the general problem B is nevertheless infeasible. This problem is one of the reasons for the limited applicability to cryptography of the theory of NP-completeness. One can show that this problem does not arise in our case because a Diffie-Hellman oracle that answers correctly for a fraction ϵ of the inputs can be transformed into a uniform Diffie-Hellman oracle.

We are only interested in groups G for which the discrete logarithm problem is believed to be intractable. The fastest algorithm for general groups, which is attributed to Shanks and referred to as the baby-step giant-step algorithm, runs in time $O(\sqrt{n} \log n)$ and requires space $O(\sqrt{n})$, where n is a known upper bound on $|G|$. If $|G|$ is known, an algorithm of Pollard with essentially the same running time but almost no space requirement can be used. However, its running time has not been proven rigorously. Furthermore, it is well-known [15] that for an arbitrary group G, discrete logarithms can be computed in time $O(\sqrt{q})$ where q is the largest prime factor of the order $|G|$ of G. For certain specific groups, such as the multiplicative group modulo p, there exist algorithms which run much faster than the generic algorithms. We refer to [12] for a detailed discussion of the discrete logarithm problem and algorithms for solving it.

The first published result on the equivalence between computing discrete logarithms and breaking the Diffie-Hellman protocol is due to den Boer [4]. He proved the equivalence for the group Z_q^* when q is a prime such that $\varphi(q-1)$ has only small prime factors. In order to avoid any confusion it should be pointed out that this is not equivalent to the condition that $q-1$ be smooth, and that no efficient discrete logarithm algorithm is known for Z_q^* for primes q of the described special form. In the following we give a generalized description of den Boer's idea which will serve as an introductory example for our proof technique.

For simplicity, assume that G is a group with prime order $|G| = p$ where

$p-1=\prod_{j=1}^{r} q_j$ with $q_j \leq B$ for all j for some smoothness bound B, where the q_j's are pairwise relatively prime. The arguments can easily be generalized to arbitrary groups with known factorization of the group order, which also need not be square-free if a DH-oracle for subgroups of G is also available. The group G can be arbitrary, for instance a subgroup of a multiplicative group modulo a larger prime (as suggested in [19] and in the recent NIST proposal for a digital signature standard), or an elliptic curve. Let $a = g^x$ be given. The case $x = 0$ is easily detected because in this case a is simply the neutral element of G. Let c be a primitive element of $\mathbf{F}_p$. Note that such a c can easily be found when the factorization of $p-1$ is known. If $x \neq 0$, then we have

$$x \equiv c^w \pmod{p}$$

for some w satisfying $0 \leq w < p-1$. Instead of computing x directly we will compute w by computing w modulo all the q_j and using the Chinese remainder theorem, i.e., by computing $w_1, \ldots, w_r$ where

$$w \equiv w_j \pmod{q_j}$$

and $0 \leq w_j < q_j$ for $j = 1, \ldots, r$. We have

$$w \cdot \frac{p-1}{q_j} \;\equiv\; w_j \cdot \frac{p-1}{q_j} \pmod{(p-1)}$$

and hence

$$x^{\frac{p-1}{q_j}} \;\equiv\; c^{w \cdot \frac{p-1}{q_j}} \;\equiv\; c^{w_j \cdot \frac{p-1}{q_j}} \pmod{p}.$$

Thus w_j can be determined by computing

$$g^{\left(c^{t\frac{p-1}{q_j}}\right)}$$

for $t = 0, 1, 2, \ldots$ until this group element is equal to $g^{\left(x^{\frac{p-1}{q_j}}\right)}$. The latter group element can be computed by $O(\log p)$ applications of the DH-oracle.

The total computational effort for computing x corresponds to $O(\log p)$ applications of the DH-oracle and $O(B(\log p)^2/\log B)$ group operations. For any $u < B$ (e.g., $u = \sqrt{B}$), a baby-step giant-step-type time-memory tradeoff allows to reduce the number of group operations by a factor u at the expense of increasing the number of calls to the DH-oracle by a factor u and further requiring a table of size u, which must also be sorted.

3. Towards an equivalence proof for all groups

The arguments of the previous section only apply to groups G for which every large prime factor p of $|G|$ occurs only once and is of the special form that $p-1$ is smooth with respect to a small bound B. Although no DL-algorithm is known for groups of this type that is faster than for general groups of the same size, it appears to be questionable whether it is secure to use such groups only for the benefit of being able to prove the equivalence.

In this section we present a proof technique that applies to any group G. It can be viewed as a generalization of the technique discussed in the previous section in a similar sense as the elliptic curve factoring algorithm [8] is a generalization of the $(p-1)$-factoring algorithm [16]. In other words, we exploit the fact that there exist collections of groups defined algebraically over $\mathbf{F}_p$ whose orders vary over a certain interval.

Let the order of the group G be given by

$$|G| = \prod_{i=1}^{s} p_i^{e_i}. \tag{1}$$

In the (generally unlikely) case that the square of a large prime p_i divides $|G|$ (i.e., $e_i > 1$), one needs a DH-oracle not only for the group G with generator g but also for subgroups of G of the form $\langle g^{p_i^z} \rangle$, for $z = 1, \ldots, e_i - 1$. This case will not be discussed further in this paper.

In the sequel, consider a cyclic group $G = \langle g \rangle$ and any single prime divisor p of $|G|$, i.e., let

$$|G| = p \cdot h$$

where $\gcd(p, h) = 1$. For instance, $|G|$ may be given by (1) where $p = p_i$ and $e_i = 1$ for some i. Let $a \in G$ with $a = g^x$ be given and consider the problem of computing x modulo p, i.e., computing the unique x' satisfying

$$x \equiv x' \pmod{p} \tag{2}$$

and $0 \leq x' < p$.

For any two group elements g^z and $g^{z'}$ we can test whether $z \equiv z' \pmod{|G|}$ simply by testing equality of g^z and $g^{z'}$ in G. In order to test the more general condition

$$z \equiv z' \pmod{p}$$

we note that this condition is equivalent to

$$hz \equiv hz' \pmod{|G|}$$

which is satisfied if and only if

$$(g^z)^h = g^{hz} = (g^{z'})^h = g^{hz'}. \tag{3}$$

Equality of logarithms modulo p can thus be tested by two exponentiations with exponent h and a comparison in the group.

For the purpose of illustration we describe our proof techniques by applying an elliptic curve over the field $\mathbf{F}_p$, but it can be generalized to other groups defined algebraically over $\mathbf{F}_p$ such as the multiplicative group of an extension field or certain subgroups thereof, elliptic curves over extension fields of $\mathbf{F}_p$ or the Jacobian of hyperelliptic curves of $\mathbf{F}_p$.

Assume now that we know the parameters A and B of a cyclic elliptic curve $E_{A,B}(\mathbf{F}_p)$ over $\mathbf{F}_p$ which is defined as the set of points $\{(x,y) \in \mathbf{F}_p \times \mathbf{F}_p : y^2 = x^3 + Ax + B\}$ together with the point $\mathcal{O}$ at infinity, where the order of the curve is given by

$$T \;:=\; \# E_{A,B}(\mathbf{F}_p) \;=\; \prod_{j=1}^{r} q_j^{f_j} \tag{4}$$

with $q_j \leq B$ for $1 \leq j \leq r$. It is well-known that

$$p - 2\sqrt{p} + 1 \;\leq\; \# E_{A,B}(\mathbf{F}_p) \;\leq\; p + 2\sqrt{p} + 1 \tag{5}$$

and that all orders in this range are taken on for some parameters A and B. Furthermore, a theorem of Rück [18] implies that for each order in this interval there exists a *cyclic* elliptic curve. We also refer to [13] for an introduction to elliptic curves.

Very little is known about the existence of smooth numbers in the interval (5) of interest for a given prime p. However, it is known [3] that for every fixed u,

$$\psi(n, n^{1/u})/n = u^{-(1+o(u))u}$$

where $\psi(n,y)$ denotes the number of integers $\leq n$ with no prime divisor $\geq y$. This fact suggests that every integer n has the property P_c defined below for some $c > 1$.

Definition 2. An integer n has *property* P_c if there exists an integer b satisfying $n - 2\sqrt{n} + 1 \leq b \leq n + 2\sqrt{n} + 1$ with no prime divisor greater than $n^{c/u}$, where u is defined by $u^{2u} = n$.

Conjecture. There exists a constant $c > 1$ such that all sufficiently large n have property P_c.

An even stronger conjecture is that the above conjecture holds for any $c > 1$. For example, let n be a 100-digit number and note that $10^{100} \approx 33^{66}$. One can therefore expect to find an integer in the interval $[n - 10^{50}, n + 10^{50}]$ with no prime divisor greater than $10^{c \cdot 100/33}$ for some c. For $c = 1.1$ and $c = 2$ this bound is approximately 2000 and 10^6, respectively.

If p has a special form for which an elliptic curve with smooth order can be constructed efficiently, then our proof technique described below allows to prove that computing x' can be reduced efficiently to breaking the Diffie-Hellman protocol. If p has no special form but has property P_c for some small c, our proof technique allows to prove that for a general group G for which p divides $|G|$ there exists a small fixed piece of information (the elliptic curve parameters) which, when given, allows to reduce the problem of computing x' to breaking the Diffie-Hellman protocol.

One can explicitly construct certain super-singular elliptic curves with known order. For example, the curves defined over $\mathbf{F}_p$ for $p \equiv 3 \pmod 4$ by the equation $y^2 = x^3 + ax$ have order $p+1$ as do the curves defined over $\mathbf{F}_p$ for $p \equiv 2 \pmod 3$ by the equation $y^2 = x^3 + b$. An alternative group for exploiting the smoothness of $p+1$ is the subgroup of order $p+1$ of $\mathbf{F}_{p^2}$. This group can also be constructed for that quarter of the primes (namely those with $p \not\equiv 1 \pmod{12}$) for which no appropriate super-singular elliptic curve can be obtained.

Let us assume for now that a cyclic elliptic curve $E = E_{A,B}(\mathbf{F}_p)$ with smooth order $T = \prod_{j=1}^{r} q_j^{f_j}$ is given and that we wish to compute x' according to (2) where $a = g^x$ in G. Let $P = (u, v)$ be a generator of E. We can consider x' to be the x-coordinate of a point on $E_{A,B}(\mathbf{F}_p)$. If there exists no such point, i.e., if $x^3 + Ax + B$ is a quadratic non-residue modulo p, then an expected number of only two random choices $d \in \mathbf{F}_p$ is required until for $x'' = x + d$, $(x'')^3 + Ax'' + B$ is a quadratic residue modulo p. Let y be one of the corresponding y-coordinates, i.e., let (x'', y) be a point on E. This point can be written as some multiple of the generator P of E:

$$(x'', y) = w \cdot P. \tag{6}$$

If we can determine w, then we can compute x'' and hence x' because we know d.

We now describe an efficient algorithm using the DH-oracle for computing w from $a = g^x$. Note first that we can compute $g^z = g^{x^3+Ax+B}$ by two applications of the DH-oracle and $O(\log A + \log B) = O(\log p)$ group operations. Here z is a quadratic residue modulo p if and only if $z^{(p-1)/2} \equiv 1 \pmod{p}$. This condition is equivalent to

$$hz^{(p-1)/2} \equiv h \pmod{|G|}$$

and thus also to

$$g^{hz^{(p-1)/2}} \;=\; \left(g^{\left(z^{(p-1)/2}\right)}\right)^h \;=\; g^h.$$

Testing quadratic residuosity of z modulo p is thus equivalent to testing equality in G of two elements of G, which can be computed from g^z by $O(\log p)$ applications of the DH-oracle and $O(\log h) = O(\log |G|)$ group operations. If z is not a quadratic residue modulo p we can perform the same check for g^{x+d} for randomly selected d's, until it is successful. For the first d for which $z = (x+d)^3 + A(x+d) + B$ is a quadratic residue modulo p we can compute g^y from g^z where $y^2 \equiv z \pmod{p}$ by using the DH-oracle in a modular square root algorithm [9, 14]. Hence $(x + d, y)$ is a point on E. This step requires $O(\log p)$ calls to the DH-oracle.

We further note that for given pairs (g^{u_1}, g^{v_1}) and (g^{u_2}, g^{v_2}), where (u_1, v_1) and (u_2, v_2) are points on E (not known explicitly), we can compute the pair (g^{u_3}, g^{v_3}) such that $(u_3, v_3) = (u_1, v_1) + (u_2, v_2)$ on E. This is achieved by using a standard algorithm for addition on elliptic curves [13], where multiplications modulo p are replaced by calls to the DH-oracle. Note that u_1, v_1, u_2 and v_2 need not be, and that u_3 and v_3 generally will not be, in reduced form modulo p. If the points on E are represented in affine coordinates as shown here, one such hidden elliptic curve addition requires $O(\log p)$ calls to the DH-oracle. However, if the points on E are represented in projective coordinates, only a constant number of oracle calls are needed. The conversion from projective to affine coordinates requires $O(\log p)$ oracle calls.

In order to compute x'' (which satisfies $x + d \equiv x'' \pmod{p}$) we compute w defined by (6) where y is such that $(x + d, y)$ is a point on E (see above). Let

w_{jk} for $1 \leq j \leq r$ and $0 \leq k < f_j$ be defined uniquely by $0 \leq w_{jk} < q_j$ and

$$w \equiv \sum_{k=0}^{f_j-1} w_{jk} q_j^k \pmod{q_j^{f_j}}.$$

The number w can easily be computed from the w_{jk}'s using the Chinese remainder theorem.

Consider a specific j for which we wish to determine $w_{j0}, \ldots, w_{j,f_j-1}$. We have

$$\frac{T}{q_j}\, w \;\equiv\; \frac{T}{q_j}\, w_{j0} \pmod{T}$$

and therefore

$$(u', v') \;:=\; \frac{T}{q_j} \cdot (x'', y) \;=\; \left(\frac{T}{q_j}\, w\right) \cdot P \;=\; \left(\frac{T}{q_j}\, w_{j0}\right) \cdot P$$

on E. We can thus compute $(g^{u'}, g^{v'})$ from $(g^{x''}, g^{y})$ by using $O(\log p)$ oracle calls (for projective coordinates, $O(\log^2 p)$ for affine coordinates). Using "normal" group operations we can now compute $(g^{hu'}, g^{hv'})$. We further compute $(g^{hu''_t}, g^{hv''_t})$ for $t = 0, 1, 2, \ldots$, where

$$(u''_t, v''_t) = \left(\frac{T}{q_j}\, t\right) \cdot P$$

on E, using normal operations in G, and compare the pairs $(g^{hu''_t}, g^{hv''_t})$ and $(g^{hu'}, g^{hv'})$ as suggested by (3), until a match is found for some t, which is set equal to w_{j0}. Note that $t < q_j$, that is at most q_j trials are needed.

The numbers w_{jm} for $m \geq 1$ can be computed by a generalization of the described method. This allows to prove the following theorem.

Theorem 1. *Let $G = \langle g \rangle$ be an arbitrary cyclic group with order $|G| = \prod_{i=1}^{r} p_i^{e_i}$. If for each prime p_i the parameters A_i and B_i of a cyclic elliptic curve $E_{A_i,B_i}(p_i)$ with smooth order for a smoothness bound B are given, then discrete logarithms in G can be computed using $O(\log^2 |G|)$ calls to the DH-oracle and $O((B/\log B)$ $\log^2 |G|)$ group operations. If $e_i > 1$ for some i, then a DH-oracle for subgroups of G is also required.*

Corollary 2. *For groups whose order is such that for every p_i one can construct an elliptic curve according to Theorem 1 (or another cyclic group with smooth order defined algebraically over $\mathbf{F}_{p_i}$), breaking the Diffie-Hellman protocol is equivalent to computing discrete logarithms. Among these groups are those for which either $p_i - 1$ or $p_i + 1$ is smooth for all i.*

Corollary 3. *If the stated number-theoretic conjecture is true, then for every group $G = \langle g \rangle$ with known order $|G|$ there exists a side information string S of length at most $2 \log |G|$ such that when given S, breaking the Diffie-Hellman protocol for G and base g is polynomial-time equivalent to computing discrete logarithms in G to the base g. If $|G|$ contains multiple prime factors greater* *than* $(\log |G|)^k$ *for some fixed* k*, then the equivalence only holds with respect to breaking the Diffie-Hellman protocol for certain subgroups of* G*.*

Remarks.
(1) The string S consists of appropriate elliptic curve parameters for all prime divisors of $|G|$. It need not be assumed in Corollary 3 that $|G|$ be known because it can also be computed from S. Note that the order of G and its factorization are known in many proposed cryptographic applications. In fact, it is often suggested to use a group (or subgroup) of prime order.
(2) When the order of G is not known, it is conceivable that giving $|G|$ could be of some help in computing discrete logarithms in G. However, in those cases where $|G|$ is known, there seems to be no reason to believe that knowledge of S could reduce the difficulty of the discrete logarithm problem, but this has not been proved.

4. Concluding remarks

Let p be a prime factor of $|G|$. The proof technique presented in this paper applies to any cyclic group F (with generator f) with smooth order which is defined algebraically over $\mathbf{F}_p$ and whose elements are represented by vectors over $\mathbf{F}_p$, provided it is possible to determine (by an algebraic computation) explicitly an element of F when the value of one of the coordinates is fixed. The idea is to assign to this coordinate the (hidden) logarithm x implicitly given by g^x and to perform a computation using the DH-oracle to obtain the hidden values of the other coordinates. This results in the disguised version of an element P of F. The next step is to compute certain powers of f (in disguised form) using normal group operations in G, and to compare this disguised element of F with another disguised element of F obtained from P by calls to the DH-oracle. These tests allow to compute explicitly the logarithm w of P (in F) to the base f. Given w and f we can explicitly compute the group element containing x as one of the coordinates. Examples considered in this paper were the multiplicative group of $\mathbf{F}_p$ proposed initially in [4] and elliptic curves over $\mathbf{F}_p$. The equivalence holds in a strict sense (without side information S) only if the group F with smooth order can be constructed explicitly.

It is conceivable that the application of hyperelliptic curves or some higher-degree Abelian varieties could allow to remove the plausible but unproven number-theoretic assumption from Theorem 3 because the relative sizes of the corresponding intervals for the orders of the groups are much larger than for elliptic curves. This would be similar to the generalization of the Goldwasser-Kilian elliptic curve primality test [7] to hyperelliptic curves [1], which allowed to settle the last unproven details in [7] and resulted in the first rigorously proven polynomial-time primality test.

Corollary 2 implies that if one could explicitly construct elliptic curves with smooth order for a given prime, then breaking the Diffie-Hellman protocol would be equivalent to computing discrete logarithms for all groups, without side information S. However, because the solution of the same problem for *composite* moduli would immediately yield an efficient factoring algorithm based on [8], it appears quite unlikely that this problem can be solved efficiently for prime moduli.

The results of this paper suggest to construct groups of prime order p for use in the Diffie-Hellman protocol in a manner that an explicit group (defined modulo p) with smooth order is known. Note that the description of the group need not be published but only built into the system design in a secret manner. However, it appears questionable whether using a group G with prime order $|G| = p$ such that $p-1$ or $p+1$ is smooth is a good idea, although no efficient discrete logarithm algorithms are known in this case. To find such an algorithm is suggested as an open problem.

Acknowledgment

I would like to thank Jim Hafner, Jürg Kramer, Kevin McCurley, Alfred Menezes, and Carl Pomerance for answering my questions on the state of knowledge in number theory about the existence of smooth numbers in certain intervals, and for other helpful comments.

References

1. L.M. Adleman and M.A. Huang, Primality testing and abelian varieties over finite fields, *Lecture Notes in Mathematics*, vol. 1512, Springer Verlag, 1992.
2. J. Buchmann and H.C. Williams, A key-exchange system based on imaginary quadratic fields, *Journal of Cryptology*, vol. 1, no. 2, pp. 107-118, 1988.
3. E.R. Canfield, P. Erdös and C. Pomerance, On a problem of Oppenheim concerning "Factorisatio Numerorum", *J. Number Theory*, vol. 17, pp. 1-28, 1983.
4. B. den Boer, Diffie-Hellman is as strong as discrete log for certain primes, *Advances in Cryptology – CRYPTO '88*, Lecture Notes in Computer Science, vol. 403, pp. 530-539, Berlin: Springer-Verlag, 1989.
5. W. Diffie and M.E. Hellman, New directions in cryptography, *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644-654, 1976.
6. T. El-Gamal, A public key cryptosystem and a signature scheme based on the discrete logarithm, *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469-472, 1985.
7. S. Goldwasser and J. Kilian, Almost all primes can be quickly certified, *Proc. of the 18th Annual ACM Symposium on the Theory of Computing*, pp. 316-329, 1986.
8. H.W. Lenstra, Jr., Factoring integers with elliptic curves, *Annals of Mathematics*, vol. 126, pp. 649-673, 1987.
9. J.L. Massey, Advanced Technology Seminars Short Course Notes, Zurich, 1993, pp 6.66-6.68.
10. U.M. Maurer and Y. Yacobi, Non-interactive public-key cryptography, *Advances in Cryptology - EUROCRYPT '91*, Lecture Notes in Computer Science, Berlin: Springer-Verlag, vol. 547, pp. 498-507, 1991.
11. K.S. McCurley, A key distribution system equivalent to factoring, *Journal of Cryptology*, vol. 1, no. 2, pp. 95-105.

12. K.S. McCurley, The discrete logarithm problem, in *Cryptology and computational number theory*, C. Pomerance (ed.), Proc. of Symp. in Applied Math., vol. 42, pp. 49-74, American Mathematical Society, 1990.
13. A. Menezes, Elliptic curve public key cryptosystems, Kluwer Academic Publishers, 1993.
14. R. Peralta, A simple and fast probabilistic algorithm for computing square roots modulo a prime number, *IEEE Trans. on Information Theory*, vol. 32, no. 6, pp. 846-847, 1986.
15. S.C. Pohlig and M.E. Hellman, An improved algorithm for computing logarithms over $GF(p)$ and its cryptographic significance, *IEEE Transactions on Information Theory*, vol. 24, no. 1, pp. 106-110, 1978.
16. J.M. Pollard, Theorems on factorization and primality testing, *Proceedings of the Cambridge Philosophical Society*, vol. 76, pp. 521-528, 1974.
17. R.L. Rivest, A. Shamir, and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM*, vol. 21, no. 2, pp. 120-126, 1978.
18. H. Rück, A note on elliptic curves over finite fields, *Math. Comp.*, vol. 49, pp. 301-304, 1987.
19. C.P. Schnorr, Efficient identification and signatures for smart cards, Advances in Cryptology – CRYPTO '89, Lecture Notes in Computer Science, vol. 435, pp. 239-252, Berlin: Springer-Verlag, 1990.

This article was processed using the LaTeX macro package with LLNCS style