# Number theoretic algorithms and cryptology

Johannes Buchmann
FB Informatik
Universität des Saarlandes

July 28, 1993

# 1 Introduction

In recent years, cryptographic research has become more and more important because of the many new application areas for cryptology. Cryptographic methods are not only used to ensure privacy of electronic communication but they are also applied to find authentication and identification protocols, for example. In addition to developping new cryptosystems researchers in the field have become interested in proving the security of those systems (see [10]).

Proving the security of a cryptographic protocol is the goal of theoretical cryptology. Mathematical models for communicating machines have been invented such as interactive Turing machines and interactive proof systems and some fundamental theorems about those models have been proved. Unfortunately, so far nobody has found any efficient and provably secure cryptosystem. The question of the existence of such systems is very closely related to the hardest open problems in complexity theory such as the P=NP problem. Even if someone were to prove that the complexity classes P and NP are distinct, this would not imply the existence of secure cryptosystems since in order to find such systems it would be necessary to actually contruct a hard language in NP and to present a polynomial time method for finding hard instances in that language. So as in complexity theory, most of the results of theoretical cryptology are reduction results. The existence of zero knowledge proof systems, for example, can be reduced to the existence of secure encryption functions. Other popular security primitives are one way functions, trap door functions, oblivious transfer etc..

When it comes to implementing those primitives, cryptographers very frequently use number theoretic methods. The security of the implementations then depends on the difficulty of solving the underlying number theoretic problem. There are many such problems which have been around for a long time and for which instances, which are currently intractable, are very easy to construct. The most famous one is the problem of finding a non trivial divisor of a composite integer. A hard instance of this problem can be constructed by multiplying two prime numbers of 100 decimal digits each.

As an example for implementing a security primitive by means of number theoretic methods we discuss in this paper the implementation of the Diffie-Hellman key exchange protocol using algebraic number fields. For quadratic fields this has already been done in [3], [4], [6], [7] and [8]. We sketch a generalization of the methods described there and we give an idea how to attack the implementation by solving the discrete logarithm problem in number fields.

## 2 The general Diffie-Hellman protocol

Let $I \subset \mathcal{Z}$ be an infinite countable index set, and $\{S_i\}_{i\in I}$ be a family of finite sets. For each $i \in I$ let $f_i : S_i \times N \rightarrow S_i$ be a function. This function can be interpreted as exponentiation in $S_i$. In this situation, the Diffie-Hellman key exchange protocol works as follows: There are two communication partners, say them Alice and Bob, as usual. Alice and Bob publicly agree on an index $i \in I$ and on an element $s \in S_i$. Then Alice secretely chooses an exponent $a \in N$, she computes $\alpha = f_i(s, a)$ and transmits that element of $S_i$ to Bob. Bob secretely chooses some exponent $b \in N$, he computes the element $\beta = f_i(s, b)$, and transmits it to Alice. Now Alice and Bob run a protocol that allows them to agree on $\gamma = f_i(s, ab)$ without revealing $a$ or $b$. They use $\gamma$ as a secret key. The *Diffie-Hellman problem* is to construct $f_i(s, ab)$ from $i$, $s$, $f_i(s, a)$, and $f_i(s, b)$ and the information that is transmitted in the protocol.

The first implementation of that scheme, which was suggested by Diffie and Hellman [9] takes for $I$ the set $P$ of prime numbers, for $p \in P$ the set $S_p$ is the multiplicative group $F_p^\times$ of the prime field of characteristic $p$. The map $f_p$ is exponentiation of elements in $F_p$. If $p$ has 200 decimal digits then, given the currently known algorithms and the current computer technology, solving this Diffie-Hellman problem is infeasable.

An obvious generalization is obtained by taking for $\{S_i\}$ a family of finite abelian groups and for $f_i$ exponentiation in those groups. Those families have to be chosen such that solving the Diffie-Problem is difficult while computing in those groups is easy. It would not, for example, be a very good idea to use the additive groups $\mathcal{Z}/m\mathcal{Z}$ since solving an equation of the form $as \equiv \alpha \mod m$ for $a$ is very simple, which implies that the Diffie-Hellman problem in those groups is trivial. On the other hand, as was shown by McCurley [14], choosing the group of primitive residues modulo a composite modulus makes the system more secure than the one based on finite fields since being able to crack this system not only implies being able to solve the Diffie-Hellman problem for finite fields but also being able to factor the modulus. Another possibility is to take a family of finite extensions of a prime field of fixed characteristic.

The fastest algorithm for solving the Diffie-Hellman problem in the groups mentioned so far is probabilistic and takes expected time $L_g(1/3;\gamma)$ where $g$ is the number of elements in the group, $\gamma < 2$ and $L_g(\delta;\gamma) = \exp((\gamma + o(1))((\log g)^\delta(\log\log g)^{1-\delta}))$. The expression $L_g$ represents a linear interpolation between polynomial algorithms and exponential algorithms since $L_g(0,\gamma) = (\log g)^{\delta+o(1)}$ and $L_g(1,\gamma) = g^{\delta+o(1)}$. The algorithm is based on the number field sieve [13], [11].

Other groups which can be used in this context are the groups of points of an elliptic curve over a finite field (see [12]). The best known algorithm for solving the corresponding Diffie-Hellman problem has exponential running time, but recently computing discrete logarithms on super singular curves has beeen reduced to the problem of computing discrete logarithms in finite fields (see [15]) and this may very well be possible for arbitrary elliptic curves.

Generalizing previous results for quadratic fields (see [7], [8]) we explain in the next section how to use algebraic number fields to implement the Diffie-Hellman key exchange.

# 3 Implementing the Diffie-Hellman key exchange using algebraic number fields

We fix a degree $n$ and a signature $(s,t)$, i.e. a pair of non negative integers $s,t$ such that $s + 2t = n$. We let $I$ be an infinite set of discriminants of algebraic number fields of degree $n$. For example, if $n = 2$ we may take for $I$ the set of all positive integers $D$ with $D \equiv 0, 1 \mod 4$. We let $\{K_D\}_{D\in I}$ be a family of algebraic number fields of degree $n$ and signature $(s,t)$, i.e. each $K_D$ has $s$ real embeddings and $2t$ non real embeddings into the fields $C$ of complex numbers.

For $D \in I$ we let $S_D$ be the set of reduced ideals of $K_D$. A fractional ideal of $K_D$ is called reduced if it contains the element 1 and if the only number in that ideal all of whose archimedian values are less than 1 is the element 0. The algorithmic theory of reduced ideals is explained in [1]. Reduced ideals are small representatives of the ideal classes of $K_D$. More precisely, they can be represented with $(\log|D|)^{O(1)}$ bits. This is possible since ideals in $K_D$ are free $\mathcal{Z}$-modules of rank $n$ and therefore, they can be represented by a transformation matrix with respect to an integral basis of $K_D$. If this integral basis is chosen appropriately then the size of the transformation is polynomial in $\log |D|$. An important reason for the security of the system lies in the fact that computing a generator for the reduced ideal, when it is given by a transformation matrix, is very difficult. The set $S_D$ is not closed under ideal multiplication and therefore, $S_D$ is not even a semi group, but $S_D$ is a subset of the group $I_D$ of fractional ideals of $K_D$.

In order to define the map $f_D$ we need the following definiton of distance between two principal ideals. Fix $D \in I$. Let $||_i$, $1 \leq i \leq s+t$ be the archimedian valuations on $K_D$. For $\alpha \in K$ let $\operatorname{Log} \alpha = (\log |\alpha|_1, \ldots, \log |\alpha|_{s+t})$ be the logarithm vector of $\alpha$, Then the distance of two fractional ideals $I_1$ and $I_2$ in the same ideal class is the minimum value $||\operatorname{Log} \alpha||$ where $\alpha$ is a generator of $I_1(I_2)^{-1}$. For a reduced ideal $I$ and a positive integer $k$ we define the value of $f_D(I,k)$ to be is the reduced ideal $I'$ that is closest to $I^k$. It can be shown that the function $f_D$ can be evaluated in polynomial time (in $\log |D|$ for fixed $n$).

The Diffie-Hellman key exchange is now implemented as follows: Bob and Alice agree on a field $K = K_D$, and on a reduced ideal $I$. Then after choosing the secret exponents $a$ and $b$ they individually compute the value $I_a = f_D(I,a)$ and $I_b = f_D(I,b)$. In order to enable the communication partner to find the secret key, Bob and Alice not only transmit the values $I_a$ and $I_b$ but also the distance between $I_a$ and $I^a$ and $I_b$ and $I^b$. Then it is possible for Bob and Alice to determine the secret key $f(I,ab)$ in polynomial time.

The interesting point in this construction is that we use an infinite group to implement Diffie-Hellman. This is possible since each element in the group can be approximated in polynomial time by a "small" element of a finite subset of the group.

# 4 Cryptanalysis

We discuss how the system sketched in the previous section can be attacked. Let $K$ be an algebraic number field of discriminant $D$. Let $I$ be a reduced ideal in $K$ and suppose that for a positive integer $a$ we are given $I_a = f_D(I,a)$ and the distance of $I_a$ and $I^a$. The problem of finding $a$ is called the discrete logarithm problem in $K$.

We claim that one can solve this discrete logarithm problem if one can solve the following three problems:

1. Find the class number $h$ and a basis $e_1, \ldots, e_{s+t-1}$ for the unit lattice $\operatorname{Log} \mathcal{O}^\times$ where $\mathcal{O}$ is the maximal order of $K$.

2. For two ideals $I$ and $J$ decide whether the ideal class of $J$ is a power of the ideal class of $I$ and if so find the least positive exponent $j$ such that $I^j$ and $J$ represent the same ideal class. This is the discrete logarithm problem in the class group of $K$.

3. For an ideal $I$ decide whether it is principal and if so to find the shortest logarithm vector $\operatorname{Log} \alpha$ such that $\alpha$ generates $I$.

Suppose we have algorithms for those problems. Let $I_a = \alpha I^a$. We know $I_a$, $\operatorname{Log} \alpha$ and I. We want to find a. Write $a = ha_1 + a_2$ where $a_1$ and $a_2$ are non negative integers and $0 \leq a_2 < h$. Then $a_2$ is the smallest exponent such that $I^{a_2}$ and $I_a$ are in the same ideal class. Finding $a_2$ is, therefore, an instance of problem 2. Once we know $a_2$ we determine $J = I^h$ and $J_a = I_a I^{-a_2}$. Then $J$ is a principal ideal and since $J_a = \alpha J^{a_1}$ we see that also $J_a$ is principal. Now using the algorithm for problem 3 we find for generators $\beta_a$ of $J_a$ and $\beta$ of $J$ the logarithm vectors $\operatorname{Log} \beta_a$ and $\operatorname{Log} \beta$. Then we know that $\operatorname{Log} \beta_a \equiv a_1 \operatorname{Log} \beta + \operatorname{Log} \alpha \mod \operatorname{Log} \mathcal{O}^\times$. Therefore we can find $a_1$ if we solve the linear system $x_1 e_1 + x_2 e_2 + \ldots + x_{s+t-1} e_{s+t-1} + a_1 \operatorname{Log} \beta = \operatorname{Log} \beta_a$.

Using the ideas of [2], [3] and [4] one can develop algorithms for solving the three problems mentioned above. The expected running time of those algorithms is $L_{|D|}(1/2; \gamma)$ with $\gamma < 2$.

# 5 Remarks

It can be shown that breaking the system described here is at least as hard as factoring composite integers. This was already shown in [7] for the system that only works with imaginary quadratic fields. It is possible that being able to break the system also implies being able to compute discrete logarithms in finite fields.

The advantage of the system described here over the system in [8] is that it is secure no matter whether the class number of the field is small or large. For small class numbers the principal ideal test and the computation of a logarithm vector of a generator is difficult. For large class numbers the discrete logarithm problem in the class group is difficult.

# References


[1] J. Buchmann, *Zur Komplexität der Berechnung von Einheiten und Klassenzahlen algebraischer Zahlkörper*, Habilitationsschrift, Düsseldorf 1987.

[2] J. Buchmann *A subexponential algorithm for computing class groups and regulators of number fields*, erscheint in Seminaire de Theorie des Nombres Paris 1988–1989, 27–41.

[3] J. Buchmann, S. Düllmann, H.C. Williams, *On the complexity and efficiency of a new key exchange system*, Lecture Notes in Computer Science **434** (1989), 597 - 616.

[4] J. Buchmann, S. Düllmann, *A probabilistic class group and regulator algorithm and its implementation*, erscheint in Proceedings of the Colloquium on Number Theory, Debrecen, 1989.

[5] J. Buchmann, S. Düllmann, *On the computation of discrete logarithms in class groups*, in Proceedings CRYPTO 90.

[6] J. Buchmann, R. Scheidler, H.C. Williams, *Implementation of a key exchange protocol using real quadratic fields*, Proceedings EUROCRYPT 90 Springer Lecture Notes in Computer Science **473** (1991), 98–109.

[7] J. Buchmann, H.C. Williams, *A key exchange system based on imaginary quadratic fields*, J. Cryptology **1** (1988), 107-118.

[8] J. Buchmann, H.C. Williams, *A key exchange system based on real quadratic fields*, Lecture Notes in Computer Science **435** (1989), 335 -343.

[9] W.Diffie, M. Hellman, New directions in cryptography, IEEE Trans. Inform. Theory **22** (1976), 472–492.

[10] S. Goldwasser, *The search for provably secure crypto systems*, Manuscript, 1990.

[11] D. Gordon, *Discrete logarithms in GF(p) using the number field sieve*, mnuscript, 1990.

[12] N. Koblitz, *Elliptic curve cryptosystems*, Math. Comp. **48** (1987), 203–209.

[13] A.K. Lenstra, H.W. Lenstra Jr., M. Manasse, J.M. Pollard, *The number field sieve*, Proceedings STOC 1990, 564–572.

[14] K.S. McCurley, *A key distribution system equivalent to factoring.*

[15] A. Menezes, T. Okamoto, S. Vanstone, *Reducing elliptic curve logarithms to logarithms in a finite field*, manuscript 1990.