THE DIVIDE AND CONQUER ALGORITHM: A NEW ATTACK ON THE DISCRETE LOGARITHM PROBLEM ON ELLIPTIC CURVES AND FINITE FIELDS

Malek Jakob Kakish

Department of Computer Information Systems, Amman Arab University, P.O.Box 2234, Amman 11953, Jordan

ABSTRACT

The term data security is very essential in computer information systems, everyday huge amount of confidential and case sensitive information are sent across telecommunication networks (e.g. in sectors like military, banking, trade and telecommunication companies), or stored or used in processing systems to obtain results and help to make decisions. Such confidential information need to be protected against many kinds of threats and attacks like interception or modification which can lead to lose of money, or lose of reputation and thus destroy businesses.

To achieve high degree of security, we need algorithms that are well studied and which are proven to provide the security that we are seeking. Cryptography is the science of building cryptographic systems (e.g. RSA, ElGamal, Diffie-Hellman, Elliptic Curves cryptosystems, etc.) that allow us to encrypt and decrypt data, where the security of such cryptosystems are based on the apparent intractability of solving some mathematical number theoretic problems Such problems are generally considered as being difficult to solve if we chose the associated parameters carefully.

This attack introduces a new attack on the Discrete Logarithm Problem over elliptic curves and finite fields, this attacks is more significant elliptic curves cryptosystems because not many attacks are known on elliptic curves.

Keywords: *Crypto Analysis, Elliptic Curves, Discrete Logarithm Problem, Public Key Cryptosystems, Finite Fields*

1. INTRODUCTION

The Internet, computer and communications technology are one of the fastest technology in today's Information Society, thus it is very important to have suitable security technologies and security systems that meet all security need and requirements on this Information Society.

Many standards organizations (e.g. Internet Architecture Board (IAB), Internet Engineering Task Force (IETF), etc.) did specify a huge set of security protocols, algorithms and applications that provide security services which meets that needs for data privacy and secure communication.

Although not all users needs and wishes can be achieved in one single mechanism; however, we can note that Cryptography underlies many of the security mechanisms. Cryptographic techniques or generally Cryptography is the science of data encryption and decryption.

Cryptography [1] enables us to securely store sensitive information or transmit it across insecure networks so that it cannot be read by anyone except the intended recipient. By using a powerful tool such as encryption we gain privacy, authenticity, integrity, and limited access to data.

Cryptographic systems can be divided in private key cryptography (also known as conventional cryptography systems) and public key cryptography.

Private Key cryptography, also known as secret-key or symmetric-key encryption, has an old history, and is based on using one key for encryption and decryption. In the 1960s many modern private key cryptographic systems where developed based on Feistel cipher, e.g. Data Encryption Standard (DES), Triple Data Encryption standards (3DES), Advanced Encryption Standard (AES), The International Data Encryption Algorithm (IDEA), Blowfish, RC5, CAST, etc.

In 1976 Diffie and Hellman [2] published a revolutionary concept of public-key cryptography based on two keys (Public and Private key) that solved many weaknesses and problems in private key cryptography. Upon this, many public key cryptographic systems were invented (e.g. RSA [3], ElGamal [4], Diffie-Hellman key exchange [2], elliptic curves [5], etc.). The security of such Public key cryptosystems often based on apparently difficult mathematical number theory problems ("one way functions") like the discrete logarithm problem over finite fields and over elliptic curves, the integer factorization problem or the Diffie-Hellman Problem, etc. For more information about Cryptography History see [1].

This paper discuss a new attack on the discrete logarithm problem, the significance of this algorithm is that its time consumption is comparable to well known cryptanalysis systems and its idea can easily be adopted with other algorithm to define new algorithms and improve the performance of cryptanalysis systems to solve the discrete logarithm problem, additionally it can also easily adopted to break cryptosystems based on the discrete logarithm over elliptic curves.

2. PROBLEM FORMULATION

Let p be a prime number, then Z_p denotes the set of integers {0, 1, 2, … , p - 1}, where addition and multiplication are performed modulo p. It is well-known that there exists a non-zero element $g \in Z_p$ such that each non-zero element in Z_p can be written as a power of g such an element g is called a generator of Z_p . A group is called cyclic if such an element g exists.

A Field is a nonempty set F of elements with two operations "+" (called addition) and " ▪ " (called multiplication) satisfying the following axioms: for all a, b, c ∈F,

i. F is closed under + and ▪ , i.e., a + b and a ▪ b are in F;
ii. Commutative laws: a + b = b + a, a ▪ b = b ▪ a;
iii. Associative laws: (a + b) + c = a + (b + c), a ▪ (b ▪ c) = (a ▪ b) ▪ c;
iv. Distributive law: a ▪ (b + c) = a ▪ b + a ▪ c.

Furthermore, two distinct identity elements 0 and 1 (called the additive and multiplicative identities, respectively) must exist in F satisfying:

v. a + 0 = a for all a ∈ F;
vi. a ▪ 1 = a and a ▪ 0 = 0 for all a ∈ F;
vii. For any a in F, there exists an additive inverse element (-a) in F such that a + (-a) = 0;
viii. For any a ≠ 0 in F, there exists a multiplicative inverse element a^{-1} in F such that $a \cdot a^{-1} = 1$.

A Finite field of prime order p or prime power q = p^f (f >=1) is commonly denoted Fq or GF(q) (for Galois field) and because Z_m is a field if and only if m is a prime, we denote the field Z_m by F_m. This is called a prime field.

For simplicity I will only consider the Discrete Logarithm Problem in Zp, which can be defined as follows: If we assume Zp is a finite cyclic group of order p, where g a generator of Zp, and y ∈ Zp, Then the discrete logarithm of y to the base g, denoted $\log_g$ y, is the unique integer x, $0 \leq x \leq p-1$, such that $y = g^x$ [6]

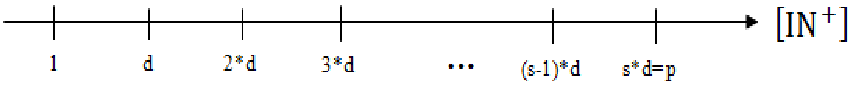
The Discrete Logarithm Problem (DLP) in Zp has been studied many years but a general solution till now was not found thus it is considered as being difficult if field's parameters are carefully chosen. In particular, there is no known polynomial-time algorithm for solving the DLP. We can classify the algorithms of solving the discrete logarithm problem in two classes: First, algorithms [7] that works only for special prime p e.g. Silver Pohlig Hellman [8], and second, general-purpose algorithms where the running times depend only on the size of p, for example exhaustive search, Shank's Baby step Giant step [9], Pollard's rho algorithm [10], Index Calculus method [11], [12] , Number field Sieve [13], etc.. The fastest general-purpose algorithms known for solving the discrete logarithm problem over finite fields are based on a method called the index-calculus.

To thwart these attacks, p should have at least 150 digits, and p - 1 should have at least one "large" prime factor, but it is important to notice that security is relative and what is secure today may need to be changed tomorrow. The use of the discrete logarithm problem in a cryptographic systems is because finding the discrete logs is (probably) difficult, but the inverse operation of exponentiation can be computed efficiently (e.g. the square-and-multiply method), this property is known in number theory as trapdoor or one-way function, there is no proof about the existence of such one way functions, but it is widely believed.

3. THE DIVIDE AND CONQUER ALGORITHM

The new attack introduced in this paper can computes Discrete Logarithm Problem in arbitrary finite cyclic group, but for simplicity I will only consider the discrete logarithm problem in Z_p and over elliptic curves.

The Divide and Conquer algorithm for finding the discrete logarithm x in finite fields Z_p (where g generator of a cyclic group Z_p of order p, $g^x = y$, y ∈ Zp) is based on the following observation:
First we divide the whole range {1,2,.., p} into small ranges of same length d:



When selecting a suitable d, we have to take the following into consideration:

(a) The smaller we select d the bigger is the d list $\{g^0, g^d, g^{2*d}, g^{3*d}, \ldots, g^{(s-1)*d}\}$

(b) The smaller we select d the faster we will find the discrete logarithm.

The worst case to find the discrete logarithm problem in the above described algorithm is to go through all numbers in the d interval; this occurs if the discrete logarithm is at the begin of the d interval.

And in average we need d/2 numbers to find the discrete logarithm.

<u>Algorithm:</u> Divide and Conquer algorithm for computing discrete logarithms x in finite fields Zp.
INPUT: A generator g of a cyclic group G of order p and an element $y \in G$ where $y \equiv g^x$ mod p.
OUTPUT: The discrete logarithm x

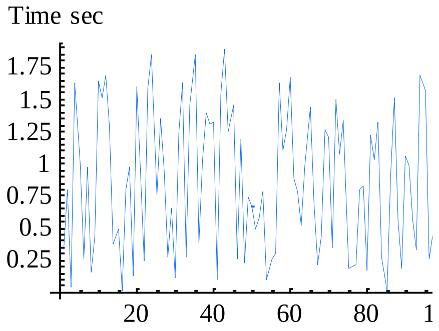
```
1. $d = \lceil p/s \rceil$, where s=$10^r$, and r $\in$ {1, 2… n} where p and r can be suitably selected.
2. Construct a table t = {{$g^0$, 0}, {$g^d$, d}, {$g^{2*d}$, 2*d}… {$g^{(s-1)*d}$ , (s-1)*d}} (Where s*d $\approx$ p) and sort by first component.
3. For w = 0 to w <=d-1 do the following:
   3.1. Use binary search, check if y equals the first component in the t table.
   3.2. If y == $g^j$ then return ( x = j – w )
   3.3. Else set y = y * $g^1$ continue at step 3
```

The running time (worst case estimation): This algorithm requires storage for O(s) group elements. The table takes O(s) multiplications to construct, and takes O(s lg p) comparisons to sort.

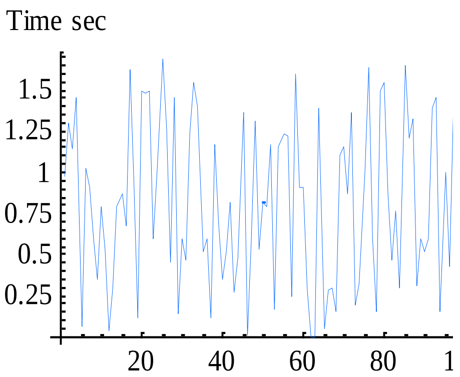
Step 3 takes O(d) multiplications and O(ld(s)) table look-ups. Under the assumption that a group multiplication takes more time than O(s lg p) comparisons, the running time of the algorithm is O(d) group multiplications.

Examples:

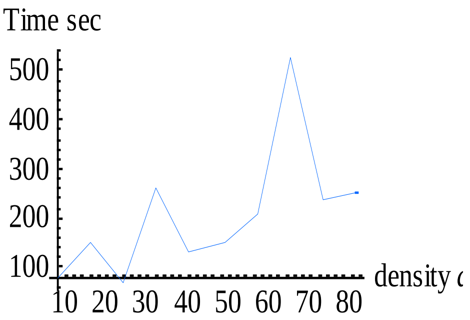
(1): The following sketch was made with Mathematica 5, it shows the solution of 100 discrete logarithm problem $y = g^x$ mod p where g = 3, p = 897579869; using Shank's baby step giant step algorithm



(2): The following sketch was made with Mathematica 5, it shows the solution of 100 discrete logarithm problem $y = g^x$ mod p where g = 3, p = 897579869;using the Divide and Conquer algorithm



Further we invest the selection of elements density d and the time it takes the Divide and Conquer algorithm to solve the discrete logarithm $y = g^x \bmod p$ where $g = 3$, $p = (2\text{^}31) -1$, the following sketch shows the solution of 50 discrete logarithms with a variable d:



This curve is maybe not exactly what we expected because it is not smooth, the main reason is because we are using random discrete logarithm numbers, but in general it shows that the bigger we choose d the longer it takes to find the discrete logarithm.

To improve the performance of Divide and Conquer algorithm we can choose s equal to Mersenne-number $M_n = 2^n - 1$[15].

A Mersenne List L_n (n>1) of length 2^n-1 of sorted elements have the following properties:

1- Each Mersenne list has a middle element. His index is at 2^{n-1}.
2- All element right (left) from middle element build also Mersenne list L_{n-1}
3- All $(2^{n-1}) - 1$ Elements right (left) from middle element are bigger (smaller) than the middle element

The maximum number of search step in a Mersenne List L_n takes maximum n steps.

In 1987 Neal Koblitz[5] proposed the use of elliptic curves E in Cryptography. To apply the Divide and Conquer algorithm on elliptic curve [16] [17] discrete logarithm problem, we define:

An Elliptic Curve E consists of the set of points (X,Y,Z) that satisfy the following homogeneous Weierstrass equation:

$$Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3$$

Where a_i (i=1, 2, 3, 4, 5, 6) are elements of a field F and with the exception that the triple (0 ,0, 0) is not a point on E. For F we can set C (complex numbers), R (reals) or Q (rationals) or any finite field Fq we wish. The advantage of using R (reals) over E is that we can analyse calculation in that field geometrically, this will help use to understand the arithmetic of Elliptic Curves and why they call them elliptic curves.

If we set Z= 0 and substitute x = X/Z, y = Y/Z then we get the equation:

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6.$$

The above equation is called the affine Weierstrass equation. If a point P satisfy the homogeneous Weierstrass equation and the equation:

$$\frac{\partial F}{\partial X} = \frac{\partial F}{\partial Y} = \frac{\partial F}{\partial Z} = 0$$

Then we call that point singular and we call the Weierstrass equation also singular, note that singular Weierstrass equations are not of interest in the Cryptogrpaphy.
We need now a criterium that can help us to determine if a given affine Weierstrass equation singular is or not. The discreminnte Δ (field element) is such a tool, which can be defined as follow:

$$\begin{aligned}
d_2 &= a_1^2 + 4a_2 \\
d_4 &= 2a_4 + a_1a_3 \\
d_6 &= a_3^2 \\
d_8 &= a_1^2a_5 + 4a_2a_6 - a_1a_3a_4 + a_2a_3^2 - a_4^2 \\
c_4 &= d_2^2 - 24d_4 \\
\Delta &= -d_2^2d_8 - 8d_4^3 - 27d_6^2 + 9d_2d_4d_6 \\
j(E) &= c_4^3 / \Delta
\end{aligned}$$

If Δ = 0, then affine Weierstrass equation is singular, otherwise not singular [18]. We call j(E) the j-invariant of the elliptic curve E. Note that only elliptic curves E over finite fields are of interest in Cryptography.

The characteristic of a field F, often denoted char(F), is the smallest positive number n such that:

$$\underbrace{1 + 1 + \ldots + 1}_{n}$$

The field is said to have the characteristic zero it this repeated sum never reaches the additive identity.

The Arithmetic of E(F) group

Now we must differentiate between the characteristics of the underlying field F that we are using to build the group of points. The following table shows different finite fields and the corresponding elliptic curve equation classified by the characteristic.

Characteristic on F	**Elliptic curve equation**
1.Char(F) = 2, j(E) ≠ 0	$y^2 + xy = x^3 + a_2x^2 + a_6$
2.Char(F) = 2, j(E) = 0	$y^2 + a_3y = x^3 + a_4x + a_6$
3.Char(F) = 3, j(E) ≠ 0	$y^2 = x^3 + a_2x^2 + a_6$
4.Char(F) = 3, j(E) = 0	$y^2 = x^3 + a_4x + a_6$
5.Char(F) > 3	$y^2 = x^3 + a_4x + a_6$

Such an elliptic curve over a finite field F is a plane curve which consists of the points that satisfy the elliptic curve equation E along with a distinguished point at inifinity, denoted O. This set together with the "addition" rules as group operation form an abelian group, with the identity element O.

The elliptic curve discrete logarithm problem is: given points P and Q in the group, find a number x such that xP = Q; x is called the discrete logarithm of Q to the base P.

Here it is important to mention that the discrete logarithm over elliptic curve with small keys is much harder to solve than the discrete logarithm over finite fields, one of the reasons is that not all attacks on finite fields can be applied on elliptic curves, the following table shows Cryptographic key length recommendation according to NIST (National Institute of Standards and Technology):

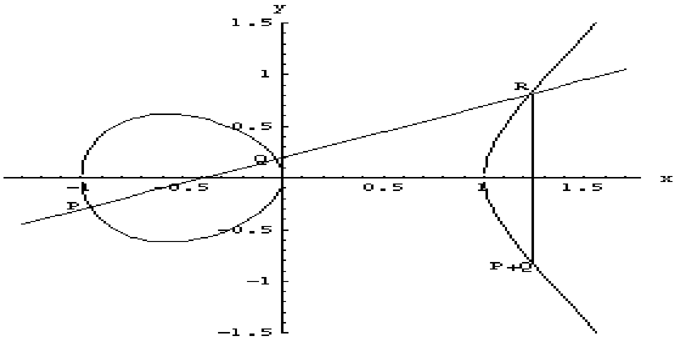
Date	**Finite fields**	**Elliptic curves**
2007-2010	1024	160
2011-2030	2048	224
>2030	3072	256
>>2030	7680	384
>>>2030	15360	512

The definition of group of points on elliptic curve E:

1. There is a point O $\in$ E, such that for all P $\in$ E, P + O = O + P = P.
2. –O = O (the identity of the group).

3. If P ≠ O and P=(x_1,y_1, then –P is (x_1,-y_1-a_1x_1-a_3) [18] .
4. If two points on E have same x-coordinate then either P=Q or P=-Q.
5. If Q = -P, then P + Q = O.
6. For two points P ≠ O and Q ≠ O on E, the addition is defined as follows. Draw the line through P and Q to intersect the curve in a third point; then reflect that point in the x-axis.
7. For two points P ≠ O and Q ≠ O on E, When P = Q, use the tangent line at P. The identity of the group is O, the "point at infinity", which conceptually lies at the top and bottom of every vertical line.

The following sketch shows the addition of two point on the elliptic curve:



Now we can describe the Divide and Conquer algorithm on elliptic curves:

__Algorithm:__ Divide and Conquer algorithm for computing discrete logarithms x over an elliptic curves E
INPUT: A base point P(x_p,y_p) of an abelian group of order m, where mP=O (identity element), and a Point Q(x_q,y_q) where xP=Q.
OUTPUT: The discrete logarithm x

1. d=m/s , where s=10^r, and r ∈ { 1, 2, …, n} where r can be suitably selected.
2. Use P(x_p,y_p) to construct the table t = {{x_p, 1}, {x_d, d}, {x_{2d}, 2d},… , {$x_{(s-1)*d}$, (s-1)*d}} (where s*d ≈ m) and sort by first component.
3. Set Q_1 = Q
4. For w = 0 to w <=d-1 do the following:

4.1. Use binary search, check if x_q equals the first component x_j in the t table.
4.2. If (x_q == x_j) and (y_q == y_j) then return (x= j-w)
4.3. If (x_q == x_j) and (y_q =/= y_j) then return (x= m – (j-w))
4.4. Else Q = Q + Q1; continue at step 3

The running time (worst case estimation): This algorithm requires storage for O(s) group elements. The table takes O(s) points addition to construct, and takes O(s lg m) comparisons to sort.

Step 4 takes O(d) points addition and O(ld(s)) table look-ups. Under the assumption that a point addition takes more time than O(s lg m) comparisons, the running time of the algorithm is O(d) points addition.

4. CONCLUSIONS

In this paper I briefly discussed the security of public-key cryptographic systems that are based on the discrete logarithm problem, however; such cryptographic system will no longer be secure if the corresponding mathematical problem is solved in the future.
The new attack described above have several advantages compared with other general purpose attacks, first it can be applied on finite fields and elliptic curves groups, this attack is of more importance on elliptic curves groups, because elliptic curves group size is much more smaller compared to finite fields groups, and where elliptic curve encryption is widely used especially on memory and processor limited devices such as smart cards, additionally the new attack running time estimation is equal to well known attacks on the discrete logarithm problem and it can easily be implemented or combined with other algorithm to solve the discrete logarithm problem. Nevertheless the

calculation of the discrete logarithm x in F_p^* where p have at least 150 digits, and p-1 have at least one "large" prime factor, will stay a hard to solve problem.
The best known algorithm for finding the discrete logarithm in finite field groups is the index calculus method, unfortunately it cannot be transformed on elliptic curves groups, the best known algorithms for any other finite abelian group (such as elliptic curves groups) is the Pollard techniques [10], Shanks Baby step Giant step algorithm [9], or the new attack described above, all these attacks are close to the best what cryptanalysis did achieve on finite fields and elliptic curves.

Many cryptographic systems (e.g. based on knapsack problem) have been broken; our assumption about the intractability of the discrete logarithm problem may change in the future due to new mathematic insights or new computer technologies [14] that may allow us to solve the Discrete Logarithm Problem in reasonable time.

5. REFERENCES

[1] D. Kahn, The Code breakers: The comprehensive History of Secret Communication from Ancient to the Internet, Published 1967

[2] W. Diffie and M. Hellman, "New directions in cryptography", IEEE Transactions on Information Theory, 22 (1976) 644-654.

[3] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public key cryptosystems. Commun. of the ACM, 21:120-126, 1978.

[4] T. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms", IEEE Transactions on Information Theory, volume 31, pages 469-472, 1985.

[5] N. Koblitz. Elliptic curve cryptosystems. Mathematics of Computation, 48:203-209, 1987

[6] A. Menezes, P. van Oorscot and S. Vanstone, Handbook of Applied Cryptography, CRC Press, ISBN: 0-8493-8523-7, 1999

[7] T. Denny, D. Weber, "The Solution of McCurley's Discrete Log Challenge", Advances in Cryptology – CRYPTO '98, Lecture Notes in Computer Science, colume 1462, Springer-Verlag, pages 458-471, 1998.

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

[9] H. Cohen. A Course in Computational Algebraic Number Theory, volume 138 of Graduate Texts in Mathematics. Springer-Verlag, 1993.

[10] J. Pollard, "Monte Carlo methods for index computation mod p", Mathematics of Computation, volume 32, pages 918-924, 1978.

[11] A. Enge and P. Gaudry, A General Framework for Subexponential Discrete Logarithm Algorithms, Manuscript 19 pp (Feb 2000)

[12] P. Gaudry, A variant of the Adleman-DeMarrais-Huang algorithm and its application to small genera, Laboratoire d' inforatique Preprint LIX/RR/99/04 (1999)

[13] D. Gordon, "Discrete logarithms in GF(p) using the number field sieve", SIAM Journal on Discrete Mathematics, volume 6, pages 124-138, 1993.

[14] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM J. Comput. 26 (1997), 1484–1509. Available at http://www.research.att.com/shor.

[15] Marin Mersenne,French mathematician, his work is stated in his Cogitata Physica-Mathematica, Paris, 1644, this Latin passage is reprinted in W. W. R. Ball, Mathematical Recreations and Essays, fifth ed., London, 1911, p. 333-334; and also in Amer. Journ. Math., v. 1, 1878, p. 235)

[16] I. F. Blake, G. Seroussi and N. P. Smart, Elliptic curves in cryptography, Cambridge Univ. Press, Cambridge, (1999).

[17] I. F. Blake, G. Seroussi and N. P. Smart, Advances in Elliptic Curve Cryptography, Cambridge Univ. Press, Cambridge, (2005).

[18] Menezes Alfred, „Elliptic Curve Public Key Cryptosystems", 1993, 4.eddition, 1997. Kluwer Academic Publishers.