On Application of Discrete Logarithms in Information – Hiding and Knapsack Problem with New Properties of S_k – Sets

A. M. H. Rizak Al-Rammahi
Communication, Department ,Najaf Engineering Technology Technical, Education, Foundation

Abstract

Discrete logarithms are applied in many cryptographic problems . For instance , in public key , and for construction of sets with distinct sums of k-elements. The purpose of this paper is to modify the method of information-hiding using discrete logarithms , introduce new properties of S_k – sets , usedthe direct product of groups to construct cyclic group and finally, present modified method for knapsack problem using each of discrete logarithms and S_k-sets.

Introduction

Let q be a prime power, then there exists a finite field GF(q). The non - zero elements of GF(q) construct a cyclic multiplicative subgroup which is generated by primitive element. For suitable notation, one can use GF*(q) =<g>, then for any element $u, u = g^k \varepsilon GF^*(q)$ is called the discrete exponentiation of basc g to the power k. The discrete logarithm of u with respect to g is that integer k, $1 \le k \le q-1$. One can write $k = \log_g^u$. The discrete logarithm of u is sometimes referred to as the index of u.

In fact , the shortest way for constructing finite cyclic group is the steps of building Galio field of prime power order. For this virtue, Galio theory was considered the corner stone in cryptosystems. For cryptographic application of Galio theory see (1) .

Let f(x) be irreducible polynomial over GF(p) modulo of degree n hence an Element h ε GF(p^n) can be regarded as polynomials h(s) over GF(p) modulo of degree less than n.

In 1976 Diffie and Hellman (2) introduced the concept of discrete logarithms Problem over finite field and therefore they added a new direction in cryptography theory. The best known algorithm for the case q = p , runs in subsequential time and was developed independently by Miller (3), Western (4), Merkle (5), Pohlig (6) and Adleman (7). A subsequential extension of Addleman's algorithm was developed by Helman (8), then the latter modified by Blake (9), Copersmith (10) and Odlyzko (11) for the case $q = p^n$ for p fixed and q growing.

The Adleman – Merkle algorithm for computing logarithms over GF(p) relies on finding small-primes and elements that factor completely into small primes (such that smooth element).

Hellman and Reneri (8) introduced the idea of virtual spanning set for computing discrete logarithm over GF(p^n), for n growing and p fixed.

Elgamal (12) modified the above two works of quadratic field as the algebriac structure.

Practically all knapsack public key cryptosystems have broken in the few years, and so essentially the only public key cryptosystemes that still have some credibility and are widely known are those whose security depends on the difficulty of factoring integers (the RSA scheme and its variants) and those whose security depend on the difficulty of computing discrete logarithms in finite fields (13).

Instead of using discrete exponentiation modulo a prime, one could possibly gain some speed by using addition on elliptic curves. For these studies – see (14) .

All of the index-calculus algorithms known for computing discrete logarithms are forms of the index-calculus algorithms (ICA) (15).In this ICA , the first stage computes the discrete logarithms of a set Q of element in the field. Second , is to obtain a number of equations in the logarithms of elements of Q. Third solved the resulting system modulo the prime factors of p-1 .On the other hand, there is a good technique for obtaining the equations – see(16,17,19,20) .

At this time, one can classify the works of solving discrete logarithms over finite field GF(q) , according to integer q , in to three types:

First , the prime field GF(p) . Second , the quadratic field GF(p^2) . Third , the field with characteristic two GF(2^n). For the above works- see (6,11,21,22,23,) , (24,25,26,27,28) , and (9,10,29,30) .

For application of discrete logarithms , Elgamal (12) used it in public key cryptography. Odlyzko (31) modified Bose - Chowla (32) method for construction of sets with distinct sums of k elements subsets.

In section (2) of this paper , mathematical concepts of building of discrete logarithms with programmed examples are discussed. The direct product method used to construct cyclic group instead of the method of finite field.

Section (3) concerned with application of discrete logarithms in steganography problem. Section (4) contains the S_k-Sets with examples. For more general case ,we introduce new properties of S_k-sets. In section (5) we show the encryption of Merkle - Hellman (33) knapsack problem, and introduce a modified encryption using S_k-sets , discrete logarithms and knapsack problem . Finally , section (6) was concerned with important comments.

2.**Discrete Logarithms**: The following are basic known definitions and theorems for studying discrete logarithms:

Definition(2.1): If K is a field extenuation of F , the element k ε K is called algebraic over F if there exist $a_0, a_1, \ldots, a_n \in F$ not all Zero , such that $a_0 + a_1 k + \ldots + a_n k^n = 0$.

In other words , k is the roots of a non zero polynomial in F[x] (the set of all polynomials where its coefficients belong to F) .

Definition(2.2): A group (G,+) is called cyclic if all elements of G can be generated from element $g \in G$. In this case it is written $G = < g >$

Theorem(2.1) (34): Let x be algebraic over F and let r(x) be an irreducible polynomial of degree n over F with x as a root. Then $f(x) = F[x]/r(x)$.

Theorem(2.2) (34): Let GF*(p^n) be the set of non zero elements in the Galio field GF(p^n). then (GF*(p^n),.)is a cyclic group of order (p^n – 1) .

Definition(2.3) : A generation g of the cyclic group (GF*(p^n) , .) is called a Primitive element of GF*(p^n) , and for notation , GF*(p^n) = <g> for proofs of above the theorems and for other details

of constructing finite field-see(34) .We now come to business of making more mathematically prices the idea that construct a multiplicative cyclic group from finite field this is a discrete exponentiation (logarithm) approach for enciphering (deciphering) systems as follows:

Procedure (2.1): (cyclic group)

1-Irreducible polynomial: Take $f(x)=x^2+m \mod P$, provided that $f(t)=0$ *for all* $t \varepsilon Z_P$.

2- Field Extension: Put $y=a+b\alpha$ where a,b ε Z_P , $\alpha^2=-m \bmod p$ and write y = (a , b).

3- Tow Operations Field: Define

(a , b) .(c , d) = (ac + α^2 bd , ad + bc) mod p.
(a , b) + (c , d) = (a + b , c + d) mod p.

4- Primitive Elements: check if $(a_0,b_0)^{p^2-2}=1$ for minimum (p^2-2) and then write $g=(a_0,b_0)$.

5- For Cyclic Group G = < g > : Compute g^x for x = 1 To (p^2-2).

6- For Discrete Logarithm:

If y is given , compute g^x (x = 2 to p^2-1), and check if g^x = y , then write x.

For illustrating the above concepts , one can take the following examples :

Example(2-1) For encoding over $GF(17^2)$ = < (1 , 3) > , $f(x)=x^2+3$ where (y_1 , y_2) is the cipher of arbitrary values of x :

x	y_1	y_2	x	y_1	y_2
1	1	3	50	13	16
10	3	11	25	1	8
251	7	13	212	10	11
226	5	7	230	8	2
246	6	11	257	9	12
283	1	7	285	1	11
288	1	0	101	13	9
173	2	1	5	10	15
44	1	2	252	9	0

Example(2-2) For decoding $y = (y_1 , y_2)$ over $GF^*(p^2) = < (a , b) >$, $f(x) = x^2 + m$; $x = \log y$; $g = (a , b)$

P	a	b	m	y_1	y_2	x
3	1	1	1	2	2	5
17	1	3	3	1	1	276
3	1	1	1	2	0	4
3	1	1	1	1	0	8
5	1	3	3	4	3	17
5	1	3	3	2	3	8
7	1	2	1	2	2	18

Example(2-3): For finding primitive elements (generators) g = (a,b) over $GF^*(p^2)$, $f(x) = x^2 + m$

P	a	b	m		P	a	b	m
3	1	1	1		11	1	4	1
5	1	1	2		13	1	3	2
7	1	2	1		17	1	3	1

Example(2- 4) : For decoding y over $GF^*(p) = < g >$, $x = \log y \bmod p$

P	g	y	x		P	g	y	x
17	3	9	2		31	3	15	21
23	3	11	7		101	2	99	51
23	3	20	13		101	2	43	42
31	3	29	9		101	2	7	9
					101	2	55	37

Example(2-5): For encoding over $GF^*(101) = < 2 >$; $y = g^x \bmod p$

x	y	x	y	x	y	x	y	X	y
1	2	14	22	97	38	50	100	10	14
2	4	15	44	98	76	51	99	11	28
3	8	16	88	99	51	52	97	12	56
4	16	17	75	100	1	53	93	13	11

Example(2-6): For computing generators g over $GF^*(p) = < g >$

P	g	P	g	P	g	P	g
3 5	2 2,3	7 11	3,5 2,6,7,8	13 17	2,6,7,11 3,5,6,7,10,11,12, 14	31 101	3,11,12,13,17 2,3,7,8,11,12

Note (2.1): One can note that when we hope to construct a cyclic group , we use the method of finite field . The difficulties and complexities are represented in finding and solving the irreducible polynomial. To overcome these difficulties and complexities, we suggest to use the following theorem for constructing a cyclic group:

Theorem (2.3)(16) :Let $G_1,G_2,\ldots,G_n$ be groups. For $(a_1,a_2,\ldots,a_n)$ and $(b_1,b_2,..,b_n)$in $\prod_{i=1}^{n} G_i$ define $(a_1,a_2,\ldots,a_n)$ $(b_1,b_2,..,b_n)$ to be $(a_1\ b_1,a_2\ b_2,\ldots,a_n\ b_n)$ Then $\prod G_i$ is a group (external direct product) .

Example (2,7) : For instance , one can take the additive cyclic group

$Z_2\times Z_3 = \langle(1,1)\rangle$

$= \{(1,1),(0,2),(1,0),(0,1),(1,2),(0,0)\}$ Where :

x	1	2	3	4	5	6
y	(1,1)	(0,2)	(1,0)	(0,1)	(1,2)	(0,0)

For more clearness , one can take a = (0,2) , and b = (1,0) . So ln a = 2 , ln b = 3 .From the other hand , ab = (1,2) and ln (a,b) = 5 which assert that ln ab = ln a + ln b is true .

3-Application of Discrete Logarithm in Steganography:

Steganography or information hiding is the art of passing information in a manner that the vary existence of the message is unknown (35) . One can summarized steganography method as follows:

1. Embedded <Data type> : something to be hidden in something else
2. Stego: the output of hiding process : something that has the embedded message hidden in it.
3. Cover: An input with an (original) form of the stego message. In some applications, such cover message is given from the outside , in others, it is during the hiding process.
4. Stego key: Additional secret data that may be needed in the hiding process. In particular , the same key is usually needed to extract the embedded message again.
5. Embedding process: The process of hidding the embedded message is called embedding process.

6. Extracting process: Getting the embedded message out of the stego message again.For other details-see (36,37) .

Following , a proposed method for information – hiding using discrete logarithms:

Procedure (3.1) (information – hiding using discrete logarithms)

1. Construct a prime field $Z_p = <g>$
2. Using discrete exponents method over Z_P for Ciphering plain text.
3. For compression data, use the code word C(p) of plain text word $P = p_1\ p_2 \ldots p_n$ such that $C(p) = n\ c_1\ c_2$ for $n \geq 3$.
4. Write the cover text.
5. Send the original text among the cover text.

Example (3.1) : For illustration, we take the following example which introduced in (38)

For encoding:

1-Take $Z_{101} = <2>$

2- compression the data : 135y is 3te 3at 3ad 7se of 13cg in a 3wy 5wh 5hs 3te 9ee of 3te 13cn In 8ct to 12cy , 5we 3te 5ey is 7ad to 6dt 9it 3ad 6my 8ms 7wt 5bn 4ae to 7ve 7cn 8sy 8ps 10gd by 12cm 3te 4gl of 13sy is to 4he 7mc 6ie 5or 8hs 7me In a 3wy 4tt 4ds 3nt 5aw 3ay 5ey to 4cn 6dt 4tt 5te is 6sd 6st 7me 7pt .

3.Cod message became:139810, 0798, 039532, 030295, 030216, 079832, 4464, 130827, 0722, 02, 035510, 055554, 055498, 039532, 093232, 4464, 039532, 130822, 0722, 080895, 9544, 120810, 055532, 039532, 053210, 0798, 070216, 9544, 061695, 090795, 030216, 061110, 081198, 075595, 050422, 040232, 9544, 077732, 070822, 089810, 088898, 102716, 0410, 120811, 039532, 042756, 4464, 139810, 0798, 9544, 045432, 071132, 060732, 054449, 085498, 071132, 0722, 02, 035510, 049595, 041698, 032295, 050255, 030210, 053210, 9544, 043222, 061695, 049595, 059532, 0798, 02, 069816, 069895, 071132, 078895.

4.Write any cover text, then send our code message among the cover text .

Note(3.1): Clearly, discrete logarithm used for decoding each code word where the first two digit represent length of plain word .

Note(3.2): There is a good available in MATLAB dealing with string text (message) numbers and the length of the word .

Note(3.3): Because of the wide area of Galio field, one can use permutation group before the assignment of discrete exponent code word .

4-New properties of S_k-sets:

Let (G,t) be abelian group , then the subset S is S_k-sets if the sums of any elements with length k selected from S are all distinct .S_k-sets was introduced by Bose and Choowla (32). They construct S_k-sets in additive group modular Z_m .S_k-sets used in combinatorial and graph theory , for instance , each of Klove (39) , and Brouwer (40) used S_k-sets in calculating lower bounds for constant weight codes. Furthermore, Chunge (41) used S_k-sets in diameter and eigenvalues.

Indeed , the good modification of Bose-Chowla construction was introduced by Odlyzko (15). In this modified work, it was showed that for each $k \geq 2$, an infinite number of values m exists to that a set S with distinct k-sums exists inside Z_m . And the notation of S_k-sets was extended to nonabelian group. They introduce a 8-elements S_3-set in Z_{156} while Bose-Chowla supplied a 6-elements S_3-set .

Following the known basic concepts of S_k-sets:

Definition (4.1): Let $S \subset Z_m$. If every non zero elements in Z_m has a unique representation a-b mod m , $a \in S, b \in S$, then S is called perfect difference set.

Definition (4.2): Let $S \subset G$,G is abelian finite group. If the sum of any k elements selected (with replacement) from S is not equal to sum of any other k element of S, then S is called S_k-set in G.

Theorem (4.1)(1): For every integer $k \geq 2$ and every prime P so that k divide P-1, there exists an S_k-set of cardinality P inside Z_m where $m = P^k - 1$,such that S = PS .

Theorem (4.2)(1): For each integer $k \geq 2$, and every prime P with k divide P-1, a non abelian group G of order $|G| = (P^k - 1).k$ exist which contains S_k-set, S of cardinality $\frac{(P-1)}{k}$.

Theorem (4.3),(1)): Let q be a prime power. Then there exists a set S of cardinality q + 2 inside the dihedral group D_{2m} of order 2m, where

$m = \frac{(q^3 - 1)}{(q - 1)}$, such that the products xy with $x, y \in S$, $x \neq y$, are distinct.(In particular, $xy \neq yx$ for $x \neq y$).

Example(4-1) The following examples are of S_k-set with cardinality P, in Z_m:

Set	k	P	m
{1,2,4};{3,5,6}	2	2	7
{1,3,4};{4,5,7}	2	3	8
{0,1,3,9};{0,12,10,4}	2	3	13
{2,5,6};{7,8,11}	2	3	13
{1,2,5,10,25,50}	3	5	124
{1,2,5,10,25,50,94,125}	3	5	156

Remark (4.1)(42): There is a b injection between any two of the following sets:

a) The set of k-letter words with distinct letters on an alphabet m letters.

b) The set of injection of an k-set into an m-set.

c) The set of distributions of k-distinct objects into m distinct boxes such that every box contains at most one object.

d) The of k- permutations of m symbols taken at a time.

The cardinality of each of these sets is $[m(m-1)\ldots(m-k+1)]$.

Remark (4.2): The multiplicative symmetry condition in Theorem (1) is necessary but not sufficient. For example one can take the set S = {0,2,6} in Z_8:

Clearly 3S = S , but S is not S_2-set because it has 0 + 0 = 0 and 2 + 6 = 0 in Z_8.

The following are new proposed properties of S_k-set.

Proposition (4.1): If S is S_k-set in abelian group G, then S^{-1} is S_k-set.

Proof : Clearly $|S^{-1}| = |S|$ since S is S_k-set , then by Theorem (1) : PS = S, and if $u \in S$ we have $pu = v \in S$ then $u^{-1} \in S^{-1}$.

To prove $pu^{-1} \in S^{-1}$

$$pu^{-1} = (up^{-1})^{-1} = (up)^{-1} = (pu)^{-1} = v^{-1} \in S^{-1}$$

Proposition (4.2): Let S be not S_k-set in Cabelian group G, then S^{-1} is not S_k-set.

Proof: Suppose $W = S^{-1}$ is S_k-set , then by proposition (1) $W^{-1} = S$ is S_k-set and that is contracting with the hypothesis .

Proposition (4.3): Let S be S_k-set , and $a \neq b, a \neq 0$, then $a + b \in S^{-1}$.

Proof: Since S is a new element, for instance if $|S| = 3$, then we have three elements, each represents a sum of two distinct elements. So

$a + b \notin S \Rightarrow c = a + b \notin S \Rightarrow Pc = Pa + Pb \Rightarrow Pc = a + b \Rightarrow Pc = c \notin S$

In the same way $= Pc^{-1} = c^{-1} \in S^{-1}$

Then each of c and c^{-1} has multiplictive symmetry with respect to P . So

$c \in S^{-1}, c^{-1} \in S$.

Proposition (4.4): S_2-set in Z_m is equivalent to perfect difference set.

Proof: Let $\alpha \in Z_m$, then $\alpha = a - d; a, d \in S$

And $a - d = c - b$ iff $a + b = c + d$

That means the sum of two elements in S are distinct if the differences in S are distinct.

Example(4-1) For illustration above Properties, one can take S = {1,3,4} in Z_8 . Clearly S is S_2-set in Z_8 , and S^{-1} = {7,5,4} . Also S^{-1} is S_2-set in Z_8.

Notice $1 + 3 = 4 \in S^{-1}$; $1 + 4 = 5 \in S^{-1}$; $3 + 4 = 7 \in S^{-1}$ and in additive Z_8 ,

Where $1 - 3 = 6, 3 - 1 = 2, 1 - 4 = 5, 4 - 1 = 3, 3 - 4 = 7, 4 - 3 = 1$ are all different .

5- Merkle – Hellman Knapsack Encryption :

The area of "electronic mail" may soon be upon us : we must ensure that two properties of the current "Paper mail" system are preserved : (a) messages are private, and (b) messages can be signed (43).The development of cheap digital hardware has freed it from the design limitations of mechanical computing and brought the cost of high grade cryptographic devices down to where they can be used in such commercial applications as remote each dispenses and computer terminals

(2) .In this section, modified algorithm for encryption 0 –1 vector message is presented. Knapsack problem, discrete logarithms, and S_2-set were connected in one procedure.

First Merkle – Hellman Knapsack Encryption (44) must be presented:

Procedure (5.1) (Merkle – Hellman Knapsack)

For Encryption:

1. An integer n is fixed as a common system parameter.
2. Choose super increasing sequence $(a_1,a_2,\ldots,a_n)$ and modulus M such that
$M > a_1 + a_2 + \ldots + a_n$; and $b_1 > b_1 + b_2 + \ldots + b_{i-1}$, for i = 1 , 2 , …, n .
3. Select random integer W, $1 \le W \le M - 1$ such that g < d (W, M) = 1.
4. Compute b = W.a mod M.
5. A's public key is $(b_1,b_2,\ldots,b_n)$; A's private key is (M,W,a).
6. Message $x = (x_1,x_2,\ldots,x_n)$ is bit vector.
7. Cipher text $Y = x_1b_1 + x_2b_2 + \ldots + x_nb_n$

For Decryption:

1- Compute $d = W^{-1}.y \bmod M$
2- Compute $a = W^{-1}.b \bmod M$
3- Deduce x from $Y = x_1a_1 + x_2a_2 + \ldots + x_na_n$ as follows:

Function x = INF (y,a)
For i = n To 1 step –1
If y > = a_i Then x_i = 1 and y = y – a_i
Else x_i = 0 .

The following is a proposed procedure where S_2-set used as a super increasing set:

Procedure (5.2) (Discrete Logarithms & S_2 - Set & Knapsack):

For Encryption:

1. Construct A = S_2-set over $Z_P^* =< \alpha >; 0 \notin A$, $A = (a_1,a_2,\ldots,a_n)$.
2. Check if A is Knapsack set (super increasing).
3. Check if $(a_1,a_2,\ldots,a_n) < P$.
4. Choose W such that. gcd (W,P) = 1.
5. Compute B = WA mod P where $B = (b_1,b_2,\ldots,b_n)$.
6. Message $x = (x_1,x_2,\ldots,x_n)$.
7. Compute y = x.B mod P .
8. Compute $d = \alpha^y \bmod P$.

For Decryption:

1- Compute $y = \log_\alpha d \bmod P$.
2- Compute $Z = W^{-1} * y \bmod P$.
3- Compute $A = W^{-1} * B \bmod P$.
4- Check if A is S_2-set over Z_P^* .
5- Compute the plain message x = INF(Z,A) .

Example (5-1): For taking $Z_{59}^{*} =< 2 >$, W = 3 , n = 5 , A = (1, 2, 7, 14, 25) , then $W^{-1} = 20$, B = (3,6,21,42,16), therefore the following table represents the message word x , the discrete exponent code d , and ordinary code y .

x	d	y	Z
(0,0,0,0,1)	46	16	25
(1,0,0,0,0)	8	3	1
(1,1,0,0,0)	40	9	3
(1,0,0,0,1)	14	19	26
(1,1,0,0,1)	11	25	28
(0,0,1,0,0)	56	21	7
(0,1,0,0,0)	5	6	2
(0,1,1,0,0)	44	27	9

6-Conclusion : The powerful of discrete logarithms appears when it is used in information–hiding problem. In proposed modified stenography method , one can not need any special dictionary between sender and receiver. Furthermore, discrete logarithm give more complexity when it used in zero one knapsack problem. For more creditability, S_2-set used as a super increasing set in zero – one knapsack problem. New properties of S_k-set s were proved. All Pervious studies of cryptosystems used the method of Galio finite field for construction of cyclic group .Here we use the direct product of two cyclic groups to construct a cyclic group .

References:

1- Odlyzko, AM.(1999) Available at :amo@reescarch.att-com., pp.1-25.
2- Diffie, W. & Hellman, M,(1976) IEEE Trans. On Inf. Theory 22,pp.472- 494.
3- Miller, J.(1975) Math.Computer; Vol.29,pp. 155-172 .
4- Western, A. and Miller, J.(1986) Royal society math. Tables, vol.1.1.9.London - Cambrige Un.
5- Merkle, J.R.(1979) Secrete Authentication , and Public Key System ",Ph.D.dissertation, Elect. Eng. Dept. Stanyordun., Stanford, CA .USA .

6- Pohlig, S. & Hellman, M.(1978)IEEE Trans. Inform. Theory, Vol.IT-24,pp.106-110 7- Adleman, L.(1979) In Proc 20 th Ann. Focs Cont. October.

8- Helman, M.& Reyneri J.(1982)Presented at Crypto 82 Conf. Santa Barbara, CA,Ang. 9-Blake,I Fuji , H. and RanstoneR.,(1984) Siam J. alg. DBcr. Methods,Vol.5,pp. 276 - 285 .

10- Copersmith, D.(1984)IEEE Trans. Inf. Theory Vol. IT - 30, pp. 587 - 594 .

11- Odlyzko, A. M,(1985) pp.1-88.Available at :amo@reesearch.att-com.

12-Elgamal, T.(1985)IEEE Trans. Inform. Theory, Vol.IT-31(4): 473-4481.

13- Odlyzko, A. M.(1987)pp.1-4, Available at :amo@reesearch.att-com .

14- Miller, V.(1986)Lecture Notes in Computer Science 218, Springer- Verlay ,NY. PP.417- 426.

15- Odlyzko, A. M(1993)pp.1-10;Available at :amo@reesearch.att-com .

16- Adleman, L.(1994)Algorithm Number Theory: First luten Symp. Ants – I, and Available at math # 077, Springer .

17- Copprsmith, D. (1994) Math. Comp . O2:333-350 .

18- Kaltogen, E.(1995)Math. Comp.64 , pp.777-806 .

19- Pomerance, C. ,Smith J.W., & Tuler R.(1988)SIAM J.Comput.17:387- 403 .

20- Pollard J.M.(1978)Comp. 32:918-924.

21-Coppersith, D. ,Odlyzko A.M. & Shroeppel, R.(1986)Algorithmica 1:1-15.

22-LaMacchia B.A.&Odlyzko A.M(1986)pp.46-62,Available at <http://www. Research.att.com/amo>.

23- Gordan D.M.(1980) Siam J.Discr.Math.6:pp.124-138.

24- Koklitz N.(1987)Math-Comp. 48:203-209.

25- Lenstra, A.K. & Lenstra, H.W.(1990)Lecture Notes in Mathematics# 1554, Springer,1993.proc.22nd ACM Symp.Theory of Computing,pp.s64-572.1990

26- Pomerance ,C., Smith J.W.(1992)Experimental Math , 1: 89-94(1992).

27- Coppersmith, D.(1994)Linear Algebra Appl.,192:33-60.

28- Elgamal, T.(1985)IEEE Trans. Inform. Theory, IT-31(4)469-472.
29- Adleman ,L. and Huang M.D.A. (1994)Information and Computation .
30- David, E. CS 588,Securty and Privacy, Un. of Virginia , Computer SC., http://www.cs.virginia.edu/evans
31- Odlyzko ,A. M.(1993)PP.1-11,Available at : amo@reesearch.att-com.
32- Bose, R. C. & Choula, S.(1962) Math. Helvet , 37 (3) :141 – 147 .
33- Merkle, R..C. & Hellman, H.(1978)IEEE Inf. Th. IT-24(5)(Sept.) pp. 525-530.
34- Freleigh, J.B.(1993)Addison-Wesley Pub.,3e,pp.316-321;403-405,1982.
35- Johnson, N.F. & Jajodia, S.(1998)Lecture Notes in Computers Science, Springer, pp.273-289.
36- Hala, Z.A.,(2000),A Thesis Submitted to The Dep. Of Computer Science and Inf. Sys., Un.Technology,Iraq.
37- Pfitzmann, B,(1996,Lectures Notes in Computer, Springier,(1996) ,pp.347-350.
38- Hala B.A.W. Hilal ,M.Y. Aladdin, J.A.(2002) Journal of Eng.Tech. 21(3): Iraq .
39- Klove, T.(1981)IEEE Trans. Inf. Theory ,27(2):257- 258 .
40- Brouwer, A.E.& shearev, T.B.(1990)IEEE Trans. Inf. Th.,36(6): 1334-1380 .
41- Chung, F.R.K.(1989) J.AMS, 2(2), pp.187-196.
42- Kassab, J . N .(1995)Lectures Notes in Applied Combinatory,Dep.Appl.Sc.Tech.Un ,Iraq .
43- Rives, R.. ,Shmir , A. , & Adelman . L.(1978)Communications of the ACM , 21(2): 120-126.
44- Menezes, A., Orschot, P.V.,(1996)Handbock of Applied Cryptography ,CRC Press,pp.283-319.

مجلة ابن الهيثم للعلوم الصرفة والتطبيقية المجلد 19 (2) 2006

حول تطبيقات اللوغارتمات المتقطعة في مسائل اخفاء المعلومات وحقيبة الظهر مع صفات جديدة للمجاميع S_k

عادل محمد حسن رزاق الرماحي
قسم هندسة الأتصالات، الكلية التقنية ــ النجف ، هيئة التعليم التقني

الخلاصـــــــة

اللوغارتمات المتقطعة تطبق في عدة مسائل للتشفير وعلى سبيل المثال في المفتاح العام وفي تكوين مجاميع ذات جمع مختلف لكل k من عناصرها. الغرض من هذا البحث هو تطوير طريقة اخفاء المعلومات باستخدام اللوغارتمات المتقطعة وتقديم صفات جديـــدة للمجاميع S_k واستخدام طريقة الجلاء المباشر للزمر لتكوين زمرة دائرية واخيرا تقديم طريقة مطورة لمسالة حقيبة الظهر باستخدام كل من اللوغارتمات المتقطعة ومجاميع S_k .