NTIA REPORT 81-81

The Discrete Logarithm Public Cryptographic System

J.E. Hershey

U.S. DEPARTMENT OF COMMERCE
Malcolm Baldrige, Secretary

Bernard J. Wunder, Jr., Assistant Secretary
for Communications and Information

September 1981

TABLE OF CONTENTS

		Page
LIST OF FIGURES		iv
LIST OF TABLES		iv
ABSTRACT		1
1.	INTRODUCTION	1
2.	MATHEMATICAL PRELIMINARIES	2
	2.1 Fields	2
	2.2 Number Theoretic Operations and Functions	3
	2.3 Sequence Theory	4
	2.4 Guide to Further Study	5
	2.5 Finite Fields	5
	2.5.1 GF(p): Finite Fields of Order p	7
	2.5.2 GF(2^n): Finite Fields of Order 2^n Where $n > 1$	8
	2.6 The Art of Exponentiation	15
	2.7 The Inverse Problem - Finding Discrete Logarithms	19
3.	THE DISCRETE LOGARITHM PUBLIC CRYPTOGRAPHIC SYSTEM	28
	3.1 Introduction	28
	3.2 Example of the DLI	30
	3.3 Example of the DLP	31
	3.4 Implementation	32
	3.5 Other Security Considerations	32
	3.5.1 Computation Time Attack	32
	3.5.2 Active Transparency Attack	33
	3.5.3 ATA Countermeasure	33
	3.6 Architectural Tools	36
4.	SUMMARY AND CONCLUSIONS	38
5.	ACKNOWLEDGMENTS	39
6.	REFERENCES	39

LIST OF FIGURES

		Page
Figure 1.	The 'binary algorithm.'	16
Figure 2.	Algorithm for computing α^X (all operations are modulo $x^7 + x + 1$).	18
Figure 3.	Shift register defining the α-squaring table.	20
Figure 4.	Active transparency attack.	34

LIST OF TABLES

Table 1.	Field Operator '+'	6
Table 2.	Field Operator 'x'	6
Table 3.	Field Operator '+'	6
Table 4.	Field Operator 'x'	6
Table 5.	Field Operator '+'	6
Table 6.	Field Operator 'x'	6
Table 7.	Computation of Successive Powers of α's Modulo 19	8
Table 8.	Powers of α mod$(x^4 + x + 1)$	11
Table 9.	Powers of α mod$(x^4 + x^2 + 1)$	12
Table 10.	Powers of α mod$(x^4 + x^3 + x^2 + 1)$	13
Table 11.	Powers of α mod$(x^4 + x + 1)$	14
Table 12.	Split-Search Work Table	23
Table 13.	Split-Search Work Table	23
Table 14.	Residues of 3^q mod(127)	24
Table 15.	Residues of x^q mod$(x^7 + x + 1)$	25
Table 16.	Split-Search Work Table	27
Table 17.	Split-Search Work Table	27
Table 18.	Comparison of Work Required for Three Different Attacks	29

THE DISCRETE LOGARITHM PUBLIC CRYPTOGRAPHIC SYSTEM

J. E. Hershey*

The report is a study and primer of the discrete logarithm public key cryptographic system. Implementation and strengths and weaknesses are discussed.

Key Words: cryptography; Diffie-Hellman system; finite field logarithms; MITRE system; public key cryptography

1. INTRODUCTION

The past few years have witnessed the development of a most fascinating discipline termed public cryptography. Cryptography, the set of procedures for rendering messages unreadable except to those intended and those procedures for authenticating commands and intentions to prevent spoofing, is an age-old pursuit. Developed over thousands of years, it entered the 20th century as an art form. Necessity, the mother of invention, accentuated the development and refinement of cryptographic methods and techniques. Following passage through two world wars into the present technological age, the art changed quickly to a science. Of premier importance in marking this milestone, the evolvement from art to science, was the attempt to quantify defensive cryptanalysis, i.e., the attempt to determine the strength of a given system under specific scenarios. Only by standards can one structure cryptographic methods and responsibly select their parameters.

Until public cryptography, it was necessary for two parties, who wished to exchange messages securely, to previously exchange secret quantities usually termed 'keys' or 'keying variables.' These exchanges could not be made public and had to be effected through a secure medium such as by courier or other protected channel. Public cryptography may free us from this constraint and do so in an ingenious manner. The mechanism relies on the apparent asymmetric complexity of a set of operations and their inverses.

As might be expected, this mechanism has given rise to a different set of security concerns than beset 'classical' cryptography. The first concern is, obviously, the evaluation of the algorithm's strength, i.e., what might be termed

---

*The author is with the Institute for Telecommunication Sciences, National Telecommunications and Information Administration, U. S. Department of Commerce, Boulder, Colorado 80303.

'shortcuts' to reduced complexity of implementation of the inverse operations. Second, because correspondents are not in possession of privileged materiel (keys, authentication words or other secret items) prior to communications, there may be significant spoofing attacks possible. These vulnerabilities are naturally grouped under the heading of 'identification assurance' or 'resolution.'

In spite of the new genres of cryptographic worries, there are two very promising possibilities for usage of public key cryptographic systems. The first is the conventional use of protecting message traffic. The second is the protection of a cryptographic keying variable for another, perhaps 'conventional,' cryptographic system such as the Data Encryption Standard (DES). Some of the public key cryptographic systems are more naturally suited to one use than to the other. The two systems we will consider in this report are naturally suited to the latter usage.

The material in this report is intended to stand alone and serve as both an instructional tool and as the basis for supporting rationale for a report to be subsequently published which will present a suggested standard for encrypting DES cryptographic variables.

2. MATHEMATICAL PRELIMINARIES

Although this paper is not intended to be a principally mathematical work, it is nevertheless impossible to do justice to the cryptographic systems without incurring some elementary modern algebra, number theory, and sequence theory. This section is intended to acquaint the reader with terms and concepts used and to provide a reference to outside study. The section is thus more oriented to providing definitions rather than attempting to serve as a self-contained propaedeutic.

2.1 Fields

Consider a set of elements $E = \{e_1, e_2, \ldots\}$ which may be either finite or infinite and an operation denoted by '+.' If:

a) for any two elements, e_i, $e_j \varepsilon E$, $e_i + e_j \varepsilon E$

b) for every e_i, e_j and e_k in E, $e_i + (e_j + e_k) = (e_i + e_j) + e_k$

c) there exists one and only one element in E, e_I (the operation identity), such that for any element in E, e_i, $e_i + e_I = e_i$

d) for any e_i in E there exists one and only one element (the inverse) in E, e_j, such that $e_i + e_j = e_I$

then we have a mathematical structure termed a GROUP. If, further, for every e_i and e_j in E, $e_i + e_j = e_j + e_i$, then the group is termed commutative or 'abelian.' All groups with which we shall work will be abelian.

An example of an infinite abelian group is the integers over addition. For this group the additive identity is 0 and the inverse of integer a is simply -a. Note, incidentally, that the integers do not form a group under subtraction because $e_i - (e_j - e_k) \neq (e_i - e_j) - e_k$.

If we introduce another commutative operation, 'x,' and require that the elements of the '+' group, excepting the '+' identity element, form a group under 'x,' and further require that for every e_i, e_j and e_k in E, $e_i \times (e_j + e_k) = (e_i \times e_j) + (e_i \times e_k)$, then we have a mathematical structure termed a FIELD.

An example of a field is the set of rational numbers over addition and multiplication. The additive identity is 0. The multiplicative identity is 1. The additive inverse of a is simply -a. The multiplicative inverse of a ($a \neq 0$) is $\frac{1}{a}$.

2.2 Number Theoretic Operations and Functions

We define a set of elements, $E = \{e_1, e_2, \ldots\}$. If element $e_i = e_j + e_k \times e_\ell$, we note that e_k divides (is a factor of) $e_i - e_j$. We say that e_i is CONGRUENT to e_j modulo ('mod' for short) e_k. We denote this by the symbology $e_i \equiv e_j \bmod(e_k)$. As an example, all even (odd) integers are congruent to each other modulo 2.

The concept of relative primitivity is important. Two elements, a and b, are said to be RELATIVELY PRIME if they share no factors (excepting the multiplicative identity) in common. Thus, 6 and 35 are relatively prime even though neither number is itself a prime. A natural extension is the concept of GREATEST COMMON DIVISOR. The symbology c = (a, b) is defined over the positive integers as follows: 'Integer c is the largest integer that can be divided without remainder into both integers a and b.' If c = 1, then a and b are relatively prime.

A very important number theoretic function is the Euler totient or 'phi' function denoted by ϕ. This function when applied to a positive integer m, $\phi(m)$, gives the count of the number of integers relatively prime to m starting with 1 (which is relatively prime to all positive integers) and incrementing by 1 up to m. Thus, $\phi(6) = 2$ and $\phi(8) = 4$ for examples. For a prime, p, $\phi(p) = p - 1$. The function ϕ is said to be 'weakly multiplicative.' This means that $\phi(mn) = \phi(m)\phi(n)$ if $(m, n) = 1$, i.e., if m and n are relatively prime. One further result about ϕ is needed before it can be calculated for any integer and that is that $\phi(p^n) = p^{n-1}(p-1)$ if p is prime. Thus, to calculate ϕ for any positive integer, q, we proceed as follows:

a) Canonically decompose q into its (unique) product of powers of primes, i.e.,

$$q = p_1^{\alpha_1} p_2^{\alpha_2} \ldots p_r^{\alpha_r} \quad .$$

b) Use the fact that ϕ is a weakly multiplicative function and write

$$\phi(q) = \phi(p_1^{\alpha_1})\phi(p_2^{\alpha_2}) \ldots \phi(p_r^{r}) \quad .$$

c) Sequentially evaluate all right-hand terms using the result

$$\phi(p_k^{\alpha_k}) = p_k^{\alpha_k - 1} (p_k - 1) \quad .$$

As an example, $\phi(108) = \phi(2^2 \cdot 3^3) = \phi(2^2) \cdot \phi(3^3) = 2 \cdot (2 - 1) \cdot 3^2 \cdot (3 - 1) = 36$.

Euler showed that for any positive integers, a and m, such that (a, m) = 1, $a^{\phi(m)} \equiv 1 \bmod (m)$. An important special case of Euler's theorem obtains when m is a prime, p. For this case, $\phi(p) = p-1$ and $a^{p-1} \equiv 1 \bmod (p)$ (and, of course, p must not be a factor of a). This special case of Euler's theorem is known as Fermat's 'little theorem.'

2.3 Sequence Theory

Consider that we have a sequence of terms $\{z_1, z_2, \ldots, z_i, \ldots\}$. Further suppose that the terms are restricted to elements of the set $E = \{e_i, e_2, \ldots\}$. Further suppose that the ith term depends on only the n terms directly preceding it and does so in a linear fashion, i.e., $z_i = c_1 z_{i-1} + c_2 z_{i-2} + \ldots + c_n z_{i-n}$. Under these conditions we say that the sequence is linearly generated by an nth degree recursion.

The sequence generation can be conveniently represented by, and studied via, a polynomial. To derive the polynomial we first devise a square matrix relating the n-tuple $T_i = (z_i, z_{i-1}, \ldots, z_{i-n+1})$ to the n-tuple $T_{i-1} = (z_{i-1}, z_{i-2}, \ldots, z_{i-n})$. By inspection, $T_i = T_{i-1} M$ where

$$M = \begin{bmatrix} c_1 & & \\ c_2 & & \\ \vdots & & I_{n-1} \\ c_n & 0 \ \ldots \ 0 & \end{bmatrix}$$

where I_{n-1} is the identity matrix of dimension n-1.

We know that the Cayley-Hamilton theorem (Perlis, 1952) requires a square matrix to satisfy its own characteristic equation. (The characteristic equation will be the polynomial describing the sequence generation.) To determine the characteristic equation we take the following determinant: $|M - \lambda I| = 0$. For calculations done modulo 2, it is customary to use a polynomial derived from the polynomial in λ by the transformation $\lambda = x^{-1}$ (Golomb, 1967).

As an example, consider the sequence generated by the recursion $x_n = x_{n-1} + x_{n-3}$. Addition here is modulo 2 and $x_i \varepsilon \{0, 1\}$. The matrix M is quickly derived:

$$M = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

The characteristic equation is then

$$|M + \lambda I| = \begin{vmatrix} \lambda+1 & 1 & 0 \\ 0 & \lambda & 1 \\ 1 & 0 & \lambda \end{vmatrix} = 0$$

(Note that addition and subtraction are equivalent modulo 2.) Expanding the determinant we obtain $\lambda^3 + \lambda^2 + 1$. Making the transformation $\lambda = x^{-1}$, we obtain $x^3 + x + 1$ as the characteristic polynomial for this mod 2 recurrence of degree 3.[1]

2.4 Guide to Further Study

For further study in fields and associated mathematical structures, the reader is referred to Dean (1966) and Van derWaerden (1953). For further study in number theory the reader is referred to Beiler (1966) and LeVeque (1956). For further study in sequence theory the reader is referred to Golomb (1967) and Kautz (1965).

2.5 Finite Fields

We know that finite fields are possible if and only if the number of elements, N, is a power of a prime number, p, i.e., $N = p^n$ (Albert, 1956). As we noted earlier, a field has two binary operators usually denoted by '+' and 'x.' As an example of a finite field in which the number of elements is a prime to the first power (n = 1), we consider N = 3. Let the set of elements be {A, B, C}. The following tables, 1 and 2, define the field operators

[1] *Although we started with an equation whose right-hand side was zero, the characteristic equation, it is customary to retain and refer to the left-hand side as the characteristic polynomial.

Table 1. Field Operator '+'

+	A	B	C
A	A	B	C
B	B	C	A
C	C	A	B

Table 2. Field Operator 'x'

x	A	B	C
A	A	A	A
B	A	B	C
C	A	C	B

The element A is clearly the additive identity as A + e = e where eε{A, B, C}. The element B is the multiplicative identity and B x e = e.

Let us now make the assignment $\left\{\begin{matrix} A = 0 \\ B = 1 \\ C = 2 \end{matrix}\right\}$. Using this particular assignment we obtain Tables 3 and 4 below.

Table 3. Field Operator '+'

+	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

Table 4. Field Operator 'x'

x	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

Note that Tables 3 and 4 are simply the addition and multiplication tables for common modulo 3 arithmetic, that is, the '+' and 'x' binary operators are common modular addition and multiplication.

Now consider a field in which the number of elements is a prime to other than the first power. We consider $N = 2^2 = 4$. Let the set of elements be {A, B, C, D}. Tables 5 and 6 define the field operations.

Table 5. Field Operator '+'

+	A	B	C	D
A	A	B	C	D
B	B	A	D	C
C	C	D	A	B
D	D	C	B	A

Table 6. Field Operator 'x'

x	A	B	C	D
A	A	A	A	A
B	A	B	C	D
C	A	C	D	B
D	A	D	B	C

By inspection of the Tables 5 and 6, we easily identify elements A and B as, respectively, the additive and multiplicative identities, but we are frustrated in attempting to assign numerical values to all the elements and letting the binary operators be normal modular arithmetic. In other words, there appears to be a fundamental difference in character between a field with 3 elements and a field with 2^2 elements.

2.5.1 GF(p): Finite Fields of Order p

Introduction

The examples at the end of the preceding section have hinted at a very important first result which we now state without mathematical rigor.

> If the number of elements of a finite field is a prime to the first power, then a field, denoted by GF(p), can be constructed in which the elements are the residues (remainders) given on division over the natural numbers, including zero, by the prime p, and the binary operations are common modular addition and multiplication.

The construction of GF(p) using the above schema is straightforward:

1) The elements are the integers 0, 1, 2, ..., p-1
2) Addition is addition modulo p
3) Multiplication is multiplication modulo p.

Element inverses are easily computed:

1) The inverse under addition of element a is simply p-a.
2) The inverse under multiplication of element a ($a \neq 0$) is easily found with the aid of Fermat's 'little theorem.' This theorem states that $a^{p-1} \equiv 1(p)$. Upon suitably factoring the left side, we obtain $a \cdot a^{p-2} \equiv 1(p)$. In this form it is immediately evident that the inverse of element a is a^{p-2}. As an example, consider $p = 7$ and $a = 5$. By Fermat's theorem, the inverse of a is $5^5 = 3125 \equiv 3$ modulo 7. This is immediately verfied by noting that $5 \cdot 3 = 15 \equiv 1(7)$.

Primitive Roots

An element, α, is said to be a PRIMITIVE root of m if the succession of powers $\alpha, \alpha^2, \ldots, \alpha^{\phi(m)}$ are all distinct when reduced modulo m. If m is a prime, p, i.e., $m = p$, then $\phi(p) = p - 1$ and the succession of powers is p - 1 long. For this case, α generates all of the nonzero field elements.

In general, a number, m, has $\phi(\phi(m))$ primitive roots (LeVeque, 1956).

Example

As an example, consider operations modulo 19. Table 7 presents the succession of powers, reduced modulo 19, of α's chosen sequentially from the set {2, 3, 4, ..., 17, 18}.

Table 7. Computation of Successive Powers of α's Modulo 19

	α^n																
n =	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
α = 2	2	4	8	16	13	7	14	9	18	17	15	11	3	6	12	5	10	1
α = 3	3	9	8	5	15	7	2	6	18	16	10	11	14	4	12	17	13	1
α = 4	4	16	7	9	17	11	6	5	1									
α = 5	5	6	11	17	9	7	16	4	1									
α = 6	6	17	7	4	5	11	9	16	1									
α = 7	7	11	1															
α = 8	8	7	18	11	12	1												
α = 9	9	5	7	6	16	11	4	17	1									
α = 10	10	5	12	6	3	11	15	17	18	9	14	7	13	16	8	4	2	1
α = 11	11	7	1															
α = 12	12	11	18	7	8	1												
α = 13	13	17	12	4	14	11	10	16	18	6	2	7	15	5	8	9	3	1
α = 14	14	6	8	17	10	7	3	4	18	5	13	11	2	9	12	16	15	1
α = 15	15	16	12	9	2	11	13	5	18	4	3	7	10	17	8	6	14	1
α = 16	16	9	11	5	4	7	17	6	1									
α = 17	17	4	11	16	6	7	5	9	1									
α = 18	18	1																

Note that there are 6 primitive roots of 19. This is in accordance with the theory as $\phi(\phi(19)) = \phi(18) = \phi(2) \cdot \phi(3^2) = 1 \cdot 3 \cdot (3 - 1) = 6$.

2.5.2 GF(2^n): Finite Fields of Order 2^n Where n > 1

Introduction

What we have found so far is that finite fields with a first power prime number of elements are frameworks within which we can easily operate using common modular arithmetic. We will now proceed to develop a practical framework for working with finite fields involving 2^n elements (n > 1).

Consider polynomials of degree n-1 of the form

$$c_{n-1}x^{n-1} + \ldots + c_1x + c_0 \qquad (1)$$

in which each of the coefficients is either a one or a zero, i.e., $c_i \varepsilon \{0, 1\}$. We define addition of two polynomials $a_{n-1}x^{n-1} + \ldots + a_1x + a_0$ and $b_{n-1}x^{n-1} + \ldots + b_1x + b_0$ as a polynomial $r_{n-1}x^{n-1} + \ldots + r_1x + r_0$ where r_i is the modulo two sum of a_i and b_i. A little thought will show that there are 2^n polynomials

possible of form (1) and that the set of all these polynomials forms an abelian (commutative) group under addition as just defined. We are thus "halfway" to a field structure and as the logical next step we ponder what will suffice for multiplication. Clearly normal multiplication of polynomials will not work for if we consider the product $(a_{n-1}x^{n-1} + \ldots + a_0)(b_{n-1}x^{n-1} + \ldots + b_0) = a_{n-1}b_{n-1}x^{2n-2} + (a_{n-1}b_{n-2} + a_{n-2}b_{n-1})\, x^{2n-3} + \ldots + a_0b_0$ we note that the product is a polynomial of degree 2n-2. If we wish to retain multiplication in the above "normal" sense we will have to manage it so that polynomials of or exceeding the nth degree will be mapped or reduced to polynomials of degree n-1 or less. It is a remarkable result that such a mapping can be accomplished through the use of modular reduction based on a 'primitive' polynomial.

Polynomials

Like their counterparts, the integers, polynomials can be factored. For example, the polynomial $x^2 + 1$ cannot be factored over the field of polynomials with coefficients from the set of real numbers because two factors of the form $(x + a)(x + b) = x^2 + (a + b)\, x + ab$ cannot be found. However, over the field of polynomials with modulo two coefficients, $x^2 + 1$ can be factored, indeed, it is a perfect square $x^2 + 1 = (x + 1)^2$. If a polynomial is factorable into a product of polynomials of smaller degree, the polynomial is said to be REDUCIBLE. Polynomials that cannot be so factored, such as $x^2 + x + 1$, are said to be IRREDUCIBLE. It is analogous to composite and prime numbers in the realm of integers. But this is where the analogy ends for there is a further dichotomization to the set of irreducible polynomials, those irreducible polynomials that are PRIMITIVE and those that are not.

Only a polynomial, P(x), that is primitive (and this is a practical way to define primitivity) can be used as a modulus of reduction so that the set $\{0, \alpha, \alpha^2, \ldots, \alpha^{2^n-1}\}$ mod P(x) maps one-to-one onto the 2^n polynomials of form (1). The character α stands for a primitive element. One of the chief results of finite field theory is that there exists suitable P(x)'s and α's that allow construction of a finite field of 2^n elements.

Examples

I. The trinomial $x^4 + x + 1$ is primitive (and hence irreducible[2]). It is known that $\alpha = x$ is a primitive element. Thus we can start generating elements of the field by successive powers of α:

---

[2]Primitivity implies irreducibility. The converse is not true except when 2^n-1 is a (Mersenne) prime.

$\alpha = x$

$\alpha^2 = x^2$

$\alpha^3 = x^3$

$\alpha^4 = x^4 = x + 1$ Notice that $x + 1$ is the remainder obtained upon division of x^4 by the modulus $(x^4 + x + 1)$.

$\alpha^5 = x^5 = x^2 + x$

.
.
.

We will now present set A, the set $\{0, \alpha, \alpha^2, \ldots, \alpha^{15}\}$ and show that it does indeed exhibit a one-to-one mapping onto the set, set B, of all polynomials of form (1) with $m = 4$ (Table 8).

II. The polynomial $x^4 + x^2 + 1$ is reducible. In fact, it is a square: $x^4 + x^2 + 1 = (x^2 + x + 1)^2$. Because it is reducible it cannot serve as an appropriate modulus and no primitive elements exist. The behavior is interesting. Let us set $\alpha = x$ and generate elements as done in example I, above, this time reducing modulo $x^4 + x^2 + 1$. (The choice of $\alpha = x$ is, in a sense, a 'natural' choice as $\alpha = x$ will always serve as a primitive element for a primitive polynomial.) The results are displayed in Table 9. Notice that the mapping is no longer one-to-one but many-to-one.

III. The polynomial $x^4 + x^3 + x^2 + x + 1$ is irreducible but not primitive. Because it is not primitive it cannot serve as an appropriate modulus and, again, no primitive elements exist. Consider its behavior exhibited in Table 10. Note that here also we see a many-to-one mapping.

IV. Let us look once more at the behavior of the primitive trinomial $x^4 + x + 1$ considered in example I. In example I we chose $\alpha = x$ as the primitive element. Suppose we had chosen $\alpha = x^3$. Table 11 shows the results of such a choice. Notice that the mapping is many-to-one. The reason this is so is not due to reduction modulo a nonprimitive polynomial but rather due to the fallacious choice of $\alpha = x^3$ as a primitive element. A little reflection will show that $\alpha = x^3$, $\alpha^2 = (x^3)^2 = x^6$, $\alpha^3 = (x^3)^3 = x^9$, $\alpha^4 = (x^3)^4 = x^{12}$, $\alpha^5 = (x^3)^5 = x^{15}$. But $x^{15} \equiv 1$ and so $\alpha^6 = (x^3)^6 = x^{18} = x^3$. What we have discovered here is that not all nonidentity elements of a finite field are necessarily primitive. The one happy exception is the case when $2^n - 1$ is a (Mersenne) prime. For an element $\alpha = x^m$ to be a primitive element, m and $2^n - 1$ must be relatively prime (have no factors, save unity, in common).

Table 8. Powers of α mod$(x^4 + x + 1)$

Set A		Set B
Additive Identity =	0	0
α =	x	1
α^2 =	x^2	x
α^3 =	x^3	$x + 1$
α^4 =	$x + 1$	x^2
α^5 =	$x^2 + x$	$x^2 + 1$
α^6 =	$x^3 + x^2$	$x^2 + x$
α^7 =	$x^3 + x + 1$	$x^2 + x + 1$
α^8 =	$x^2 + 1$	x^3
α^9 =	$x^3 + x$	$x^3 + 1$
α^{10} =	$x^2 + x + 1$	$x^3 + x$
α^{11} =	$x^3 + x^2 + x$	$x^3 + x + 1$
α^{12} =	$x^3 + x^2 + x + 1$	$x^3 + x^2$
α^{13} =	$x^3 + x^2 + 1$	$x^3 + x^2 + 1$
α^{14} =	$x^3 + 1$	$x^3 + x^2 + x$
α^{15} =	1	$x^3 + x^2 + x + 1$

Table 9. Powers of α mod$(x^4 + x^2 + 1)$

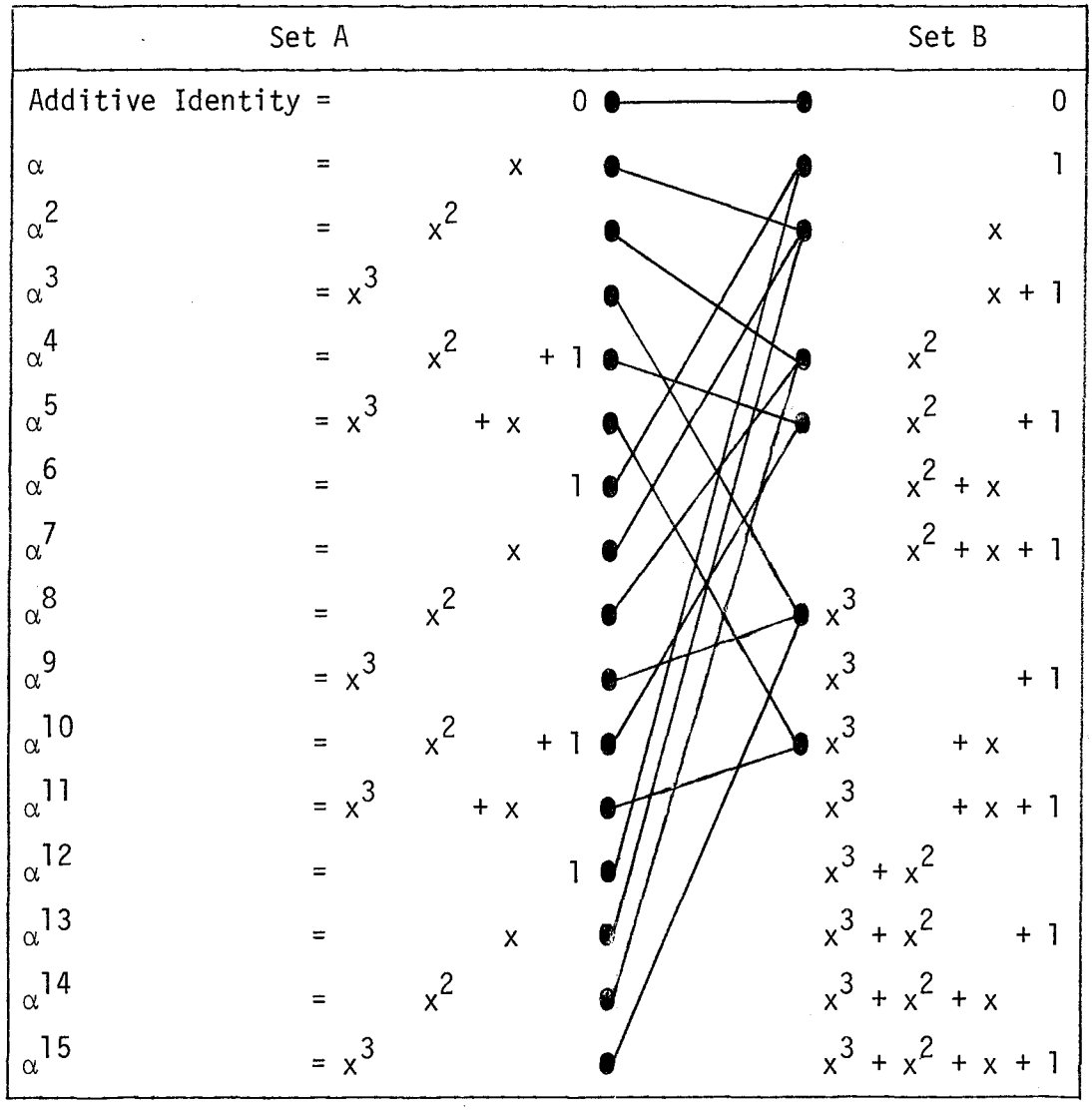

Table 10. Powers of α mod$(x^4 + x^3 + x^2 + x + 1)$

Set A		Set B
Additive Identity	= 0	0
α	= x	1
α^2	= x^2	x
α^3	= x^3	$x + 1$
α^4	= $x^3 + x^2 + x + 1$	x^2
α^5	= 1	$x^2 + 1$
α^6	= x	$x^2 + x$
α^7	= x^2	$x^2 + x + 1$
α^8	= x^3	x^3
α^9	= $x^3 + x^2 + x + 1$	$x^3 + 1$
α^{10}	= 1	$x^3 + x$
α^{11}	= x	$x^3 + x + 1$
α^{12}	= x^2	$x^3 + x^2$
α^{13}	= x^3	$x^3 + x^2 + 1$
α^{14}	= $x^3 + x^2 + x + 1$	$x^3 + x^2 + x$
α^{15}	= 1	$x^3 + x^2 + x + 1$

Table 11. Powers of α mod$(x^4 + x + 1)$

Set A		Set B
Additive Identity =	0	0
α	$= x^3$	1
α^2	$= x^3 + x^2$	x
α^3	$= x^3 + x$	$x + 1$
α^4	$= x^3 + x^2 + x + 1$	x^2
α^5	$= 1$	$x^2 + 1$
α^6	$= x^3$	$x^2 + x$
α^7	$= x^3 + x^2$	$x^2 + x + 1$
α^8	$= x^3 + x$	x^3
α^9	$= x^3 + x^2 + x + 1$	$x^3 + 1$
α^{10}	$= 1$	$x^3 + x$
α^{11}	$= x^3$	$x^3 + x + 1$
α^{12}	$= x^3 + x^2$	$x^3 + x^2$
α^{13}	$= x^3 + x$	$x^3 + x^2 + 1$
α^{14}	$= x^3 + x^2 + x + 1$	$x^3 + x^2 + x$
α^{15}	$= 1$	$x^3 + x^2 + x + 1$

2.6 The Art of Exponentiation

Given α, how many multiplications are required to obtain α^n? The answer in general is, as far as the author knows, unknown. Knuth (1969, pp. 401+) considers the problem at length. Knuth presents the following algorithm (somewhat modified by the author) which he terms the 'binary algorithm,' for accomplishing the exponentiation $Y = \alpha^n$ as shown in Figure 1.

The binary algorithm requires $\lfloor \log_2 n \rfloor + \sigma(n)$ multiplications where $\sigma(n)$ is the number of ones in n's binary representation. As stated, the algorithm is not necessarily the 'cheapest' in terms of multiplications required for general n. Knuth cites n = 15 as the smallest n for which there is a less costly procedure. The binary algorithm forms α^{15} from α with $\lfloor \log_2 15 \rfloor + \sigma(15) = 3 + 4 = 7$ multiplications. Let us, however, calculate α^{15} with only 5 multiplications as follows:

START:	$\beta \leftarrow \alpha$	
	$\beta \leftarrow \beta^2$	(1 multiplication)
	$\beta \leftarrow \beta \cdot \alpha$	(1 multiplication)
	$\gamma \leftarrow \beta$	(save α^3)
	$\beta \leftarrow \beta^2$	(1 multiplication)
	$\beta \leftarrow \beta^2$	(1 multiplication)
	$\beta \leftarrow \beta \cdot \gamma$	(1 multiplication)
FINISHED:	$\beta = \alpha^{15}$	

Although the Binary Algorithm may not always be the cheapest in terms of multiplications required, it is easily programmed and its performance, in general, is quite good. The algorithm works not only for $\alpha \varepsilon \{\text{integers}\}$ but also for $\alpha \varepsilon \{\text{finite fields}\}$.

Finally, one must bear in mind that algorithms should not always be evaluated by counting just one cost item, multiplications in this case. Total algorithmic complexity, and 'convenience,' depends upon many ancillary considerations such as storage requirements, indexing, sorting, and other housekeeping tasks.

There may also be shortcuts which do not reduce the number of multiplications, for example, but reduce the work required to perform them. As an example, the author (Hershey, 1980) has shown that the calculation of α^X for a particular set of α's in specific finite fields can be performed using a trick from the theory of recursive sequences. The method is proffered as an example following a motivational preamble.

Fast calculation of α^X:

Exponentiating α^X modulo $x^7 + x + 1$ (a known primitive polynomial) can be

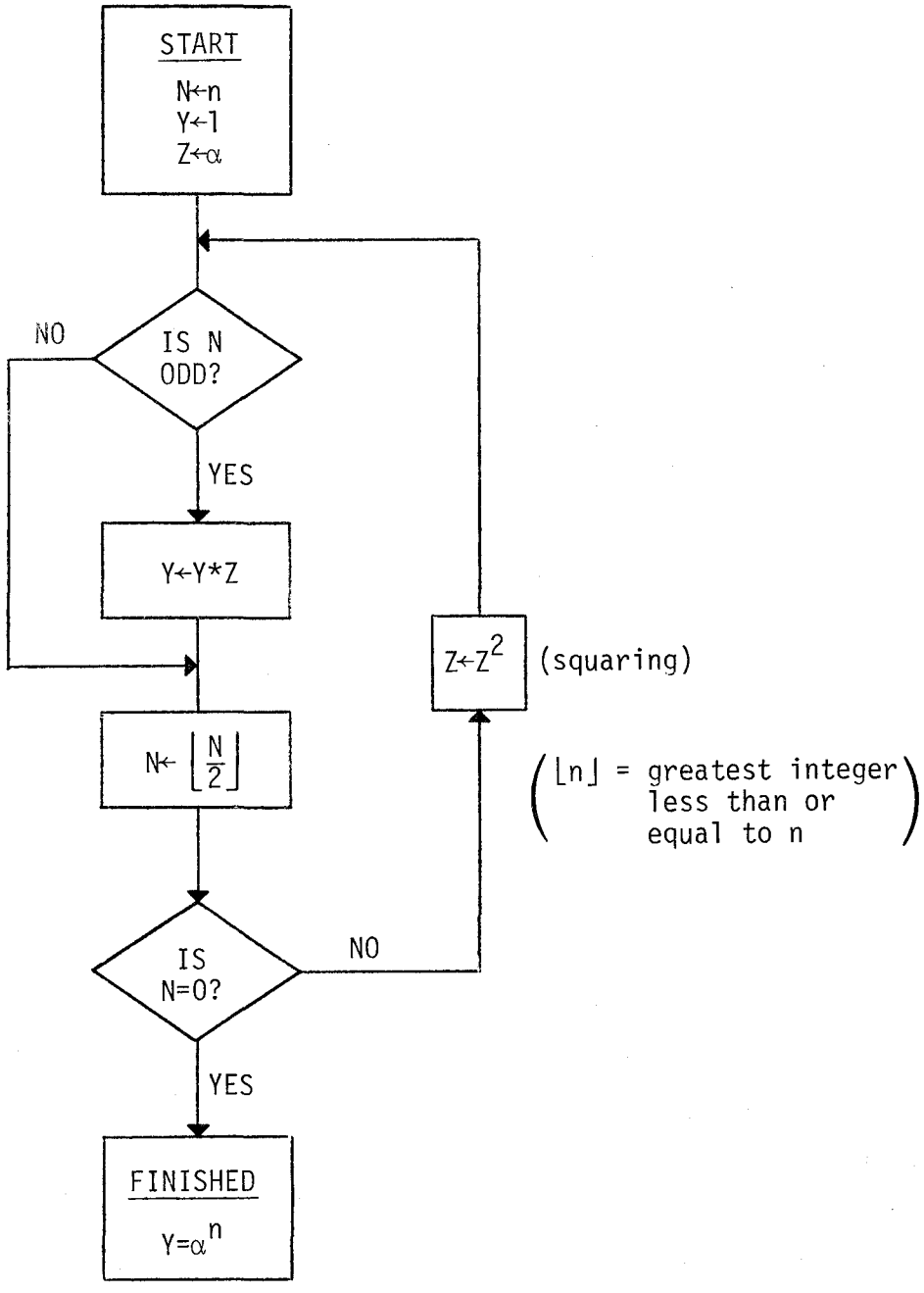

Figure 1. The 'binary algorithm.'

performed by the block structure shown in Figure 2. This structure is essentially the binary algorithm.

As an example, let our field element be $\alpha = 1 + x + x^6$ and let X = 67. These quantities are represented in a natural way by the vectors: α = (1100001), X = (1100001). The AC is a 7-bit register (vector) that serves as an accumulator. The algorithm proceeds as follows:

INITIALIZE:	AC←(1000000)
i = 1:	AC←(1000000)(1100001) = (1100001)
	α ←(1100001)(1100001) = (1010011)
i = 2:	AC←(1100001)(1010011) = (0101110)
	α ←(1010011)(1010011) = (1001011)
i = 3:	α ←(1001011)(1001011) = (1001110)
i = 4:	α ←(1001110)(1001110) = (1111101)
i = 5:	α ←(1111101)(1111101) = (1100110)
i = 6:	α ←(1100110)(1100110) = (1101100)
i = 7:	AC←(0101110)(1101100) = (0100110)
FINISHED	$\alpha^{67} = x + x^4 + x^5$

Clearly, the greatest amount of time is consumed in recursively squaring the field element α. Consider now the progression of α-vectors (α, α^2, α^4, α^8, α^{16}, α^{32}, α^{64}) when α is of the form $\alpha = c_1x + c_2x^2 + c_4x^4$, e.g., $\alpha = x$. For this case we obtain:

	1	x	x^2	x^3	x^4	x^5	x^6
α =	0	1	0	0	0	0	0
α^2 =	0	0	1	0	0	0	0
α^4 =	0	0	0	0	1	0	0
α^8 =	0	1	1	0	0	0	0
α^{16} =	0	0	1	0	1	0	0
α^{32} =	0	1	1	0	1	0	0
α^{64} =	0	1	0	0	1	0	0

Considering the table of powers of α above as a 7 x 7 array, two points are

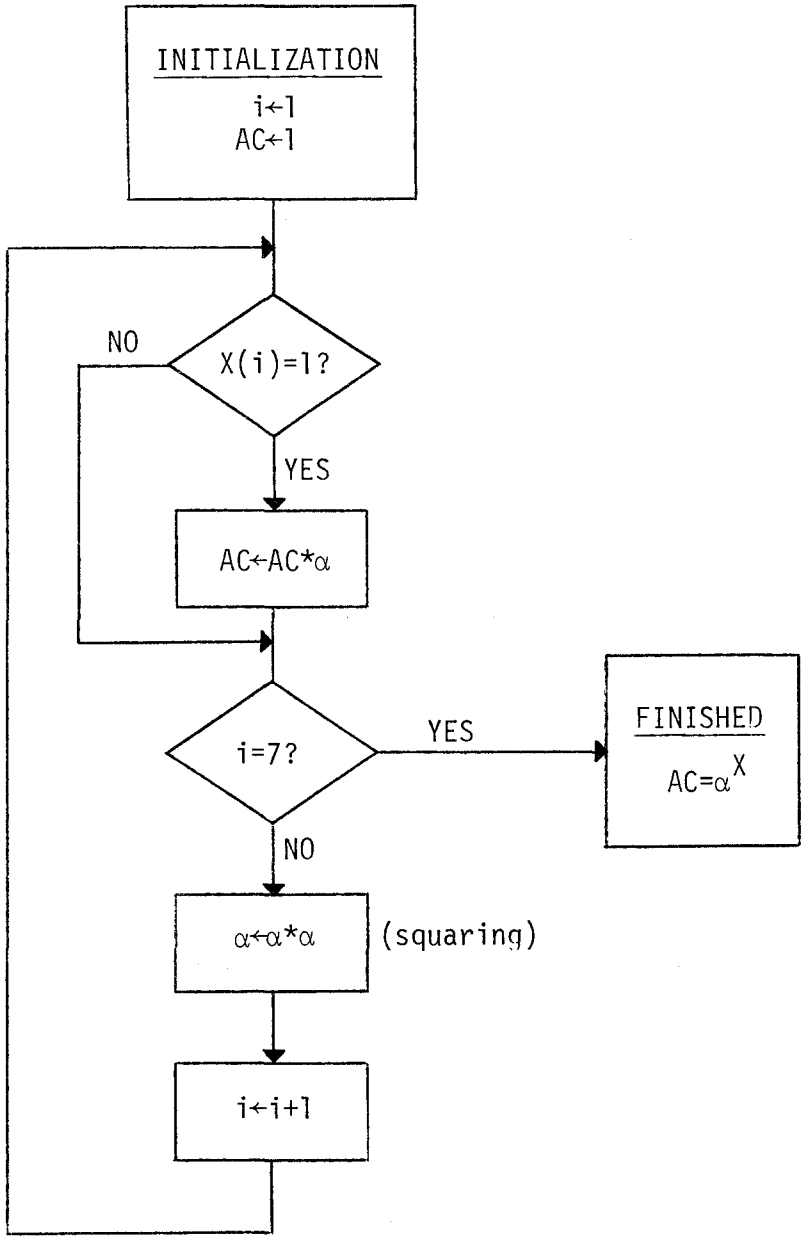

Figure 2. Algorithm for computing α^X (all operations are modulo $x^7 + x + 1$).

immediately evident:

a) It is a sparse array, there are fewer ones and zeros.

b) Many of the columns contain only zeros.

These two observations are clues to a remarkable and useful property which obtains upon choosing α to be of the form $\alpha = c_1x + c_2x^2 + c_4x^4$.

Consider that α is of the form $\alpha = c_1x + c_2x^2 + c_4x^4$. Then $\alpha^2 = c_4x + (c_1 + c_4)x^2 + c_2x^4$ and is of the same form as α. It follows that all of α^{2^i}, $i = 0, 1, \ldots, 6$, are of the same form. Also, because $x^7 + x + 1$ is primitive, all of the α^{2^i}, $i = 0, 1, \ldots, 6$, will be unique and the 'pigeon-hole' principle insures that all nonzero polynomials of the form $c_1x + c_2x^2 + c_4x^4$ will occur.

Consider, now, the bit stream of a nonzero column proceeding from top to bottom, e.g., the coefficients of x in α, α^2, α^4, α^8, α^{16}, α^{32}, α^{64}: 1 0 0 1 0 1 1. This bit stream, and hence, the entire table, is described by the smaller, one is almost tempted to term it of 'logarithmic' order, recusion: $x^3 + x^2 + 1$.[3]

To generate the successive squares of α, then, when α is of the form $c_1x + c_2x^2 + c_4x^4$, we need only implement a small shift register with linear feedback as shown in Figure 3 and step it for every squaring operation.

2.7 The Inverse Problem - Finding Discrete Logarithms

Introduction

Exponentiation, computing the forward mapping, i.e., finding α^n given α and n is straightforward and quickly accomplished. The reverse operation, the 'logarithm problem,' i.e., finding n given α and α^n is apparently not, in general, a task that can be performed in a time comparable to exponentiation. It is the apparent asymmetric complexity that provides the security of the derivative public cryptographic systems.

---

[3]The recursion generating the table for a primitive trinomial of the form $x^{2^n-1} + x + 1$ when $\alpha = c_1x + c_2x^2 + c_4x^4 + \ldots + c_{2^{n-1}}x^{2^{n-1}}$ is easily found. Note that $\alpha^2 = c_{2^{n-1}}x + (c_1 + c_{2^{n-1}})x^2 + \ldots + c_{2^{n-2}}x^{2^{n-1}}$. It is clear that the components of α, α^2, α^4... are described by a recursive process whose polynomial is $x^n + x^{n-1} + 1$.

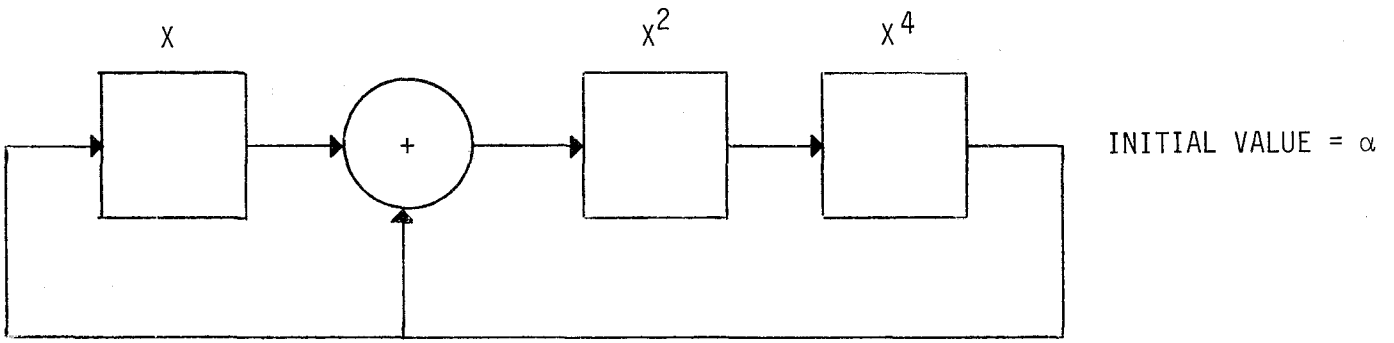

Figure 3. Shift register defining the α-squaring table.

Split-Search Attack

To find n given α and α^n we could, of course, try all possible exponents of α. The number of multiplications required on the average to find α^n then will be proportional to, or linear with, n. But we can do better. Knuth (1973, p. 9, Problem 17 [Discrete Logarithms]) presents what I call a 'split-search' algorithm which is applicable to any p.

Let

$$q = \lceil \sqrt{p} \rceil \, n_1 + n_2$$

where

$$\lceil x \rceil = \begin{cases} x \text{ if } x \varepsilon \{\text{integers}\} \\ \lfloor x \rfloor + 1 \quad \text{otherwise} \end{cases}$$

and $0 \leq n_1, n_2 < \lceil \sqrt{p} \rceil$.

It is clear that q will take on all values in the range $0 \leq q \leq p + 1$ which includes the range of interest $0 \leq q \leq p - 2$. Consider our logarithm problem to be that of recovering q given α and the equation $\alpha^q \equiv b \bmod(p)$. Substituting for q, we obtain

$$\alpha^{mn_1 + n_2} \equiv b \bmod(p)$$

where $m = \lceil \sqrt{p} \rceil$. We rewrite this equation as

$$\alpha^{mn_1} \equiv b\alpha^{-n_2} \bmod(p).$$

For simplicity of the computations that follow, we choose to rewrite further the equation as

$$\alpha^{mn_1} \equiv b(\alpha^{-1})^{n_2} \bmod(p)$$

where $\alpha^{-1} \equiv \alpha^{p-2} \bmod(p)$ by Fermat's theorem.

We now create two tables. The first table consists of $\alpha^{mn_1} \bmod(p)$ for $0 \leq n_1 < m$. The second table consists of $b(\alpha^{-1})^{n_2}$ for $0 \leq n_2 < m$. Because $q = mn_1 + n_2$ will span the entire range of possible exponents, there will be an entry in the first table that is the same as an entry in the second table. Finding these matching elements allows us to compute directly the unknown exponent q.

Example of the Split-Search Attack, using Integers

We know that $\alpha = 3$ is a primitive root of the prime 127. Therefore, for every b, $1 \leq b \leq 126$, there exists a q, $0 \leq q \leq 125$ such that $3^q \equiv b \bmod(127)$. Table 14 lists the residues of $3^q \bmod(127)$.

Let us find q such that $3^q \equiv 100 \bmod(127)$ using the split-search attack. The three steps are:

a) We calculate $m = \lceil \sqrt{127} \rceil = \lceil 11.3 \rceil = 12$

b) We calculate $\alpha^{-1} = 3^{-1} \equiv 85 \bmod(127)$

c) We prepare Tables 12 and 13.

Upon examination of Tables 12 and 13 we find that the entry 47 is common to both. Thus $n_1 = 5$ and $n_2 = 6$. We now compute $q = 12 \cdot 5 + 6 = 66$ which may be verified by looking up α^{66} in Table 14.

Example of the Split-Search Attack, using Polynomials

We know that $x^7 + x + 1$ is a primitive polynomial and that $\alpha = x$ is a primitive element. Therefore, for every polynomial of degree six or less, b(x), excepting the zero polynomial, there exists a q, $0 \leq q \leq 126$ such that $x^q \equiv b(x) \bmod(x^7 + x + 1)$. Table 15 lists the residues of $x^q \bmod(x^7 + x + 1)$.

Let us find q such that $x^q \equiv x^5 + x^3 + x + 1 \bmod(x^7 + x + 1)$. As before, $m = 12$. We calculate $\alpha^{-1} \equiv x^6 + 1$ and then Tables 16 and 17.

Upon examination of Tables 16 and 17, we find that the entry $x^4 + x^3 + 1$ is common to both. Thus $n_1 = 9$ and $n_2 = 7$. We now compute $q = 12 \cdot 9 + 7 = 115$ which may be verified by looking up α^{115} in Table 15.

2.7.3 Other Attacks

Many contemporary mathematicians are viewing the logarithm problem as one of the more interesting research areas of number theory and algebraic number theory. The problem forms a natural 'bridge' of inquiry between the two disciplines of multiplicative and additive number theory, the apparent present-day dichotomy of the 'queen of mathematical sciences.' The split-search attack was the best improvement known since Bouniakowsky's method (1870) which as Adleman (1979) points out becomes computationally equivalent to linear search as n becomes sufficiently large. The split-search attack, as we demonstrated, requires only $p^{1/2}$ multiplications but also a great deal of storage, also of the order of $p^{1/2}$, and also ancillary sorting or a special table processing architecture.

Split-Search Work Tables

Table 12. Table 13.

n_1	$3^{12n_1} \bmod(127)$	$100 \cdot 85^{n_2} \bmod(127)$	n_2
0	1	100	0
1	73	118	1
2	122	124	2
3	16	126	3
4	25	42	4
5	47	14	5
6	2	47	6
7	19	58	7
8	117	104	8
9	32	77	9
10	50	68	10
11	94	65	11

Table 14. Residues of 3^q mod(127)

q	3^q	q	3^q	q	3^q	q	3^q	q	3^q	q	3^q
0	1	21	108	42	107	63	126	84	19	105	20
1	3	22	70	43	67	64	124	85	57	106	60
2	9	23	83	44	74	65	118	86	44	107	53
3	27	24	122	45	95	66	100	87	5	108	32
4	81	25	112	46	31	67	46	88	15	109	96
5	116	26	82	47	93	68	11	89	45	110	34
6	94	27	119	48	25	69	33	90	8	111	102
7	28	28	103	49	75	70	99	91	24	112	52
8	84	29	55	50	98	71	43	92	72	113	29
9	125	30	38	51	40	72	2	93	89	114	87
10	121	31	114	52	120	73	6	94	13	115	7
11	109	32	88	53	106	74	18	95	39	116	21
12	73	33	10	54	64	75	54	96	117	117	63
13	92	34	30	55	65	76	35	97	97	118	62
14	22	35	90	56	68	77	105	98	37	119	59
15	66	36	16	57	77	78	61	99	111	120	50
16	71	37	48	58	104	79	56	100	79	121	23
17	86	38	17	59	58	80	41	101	110	122	69
18	4	39	51	60	47	81	123	102	76	123	80
19	12	40	26	61	14	82	115	103	101	124	113
20	36	41	78	62	42	83	91	104	49	125	85

Table 15. Residues of $x^q \bmod(x^7 + x + 1)$

q	x^q	q	x^q	q	x^q
0	1	22	$x^4+x^3+x^2+x$	44	x^6+x^4+x
1	x	23	$x^5+x^4+x^3+x^2$	45	x^5+x^2+x+1
2	x^2	24	$x^6+x^5+x^4+x^3$	46	$x^6+x^3+x^2+x$
3	x^3	25	$x^6+x^5+x^4+x+1$	47	$x^4+x^3+x^2+x+1$
4	x^4	26	$x^6+x^5+x^2+1$	48	$x^5+x^4+x^3+x^2+x$
5	x^5	27	x^6+x^3+1	49	$x^6+x^5+x^4+x^3+x^2$
6	x^6	28	x^4+1	50	$x^6+x^5+x^4+x^3+x+1$
7	$x+1$	29	x^5+x	51	$x^6+x^5+x^4+x^2+1$
8	x^2+x	30	x^6+x^2	52	$x^6+x^5+x^3+1$
9	x^3+x^2	31	x^3+x+1	53	x^6+x^4+1
10	x^4+x^3	32	x^4+x^2+x	54	x^5+1
11	x^5+x^4	33	$x^5+x^3+x^2$	55	x^6+x
12	x^6+x^5	34	$x^6+x^4+x^3$	56	x^2+x+1
13	x^6+x+1	35	x^5+x^4+x+1	57	x^3+x^2+x
14	x^2+1	36	$x^6+x^5+x^2+x$	58	$x^4+x^3+x^2$
15	x^3+x	37	$x^6+x^3+x^2+x+1$	59	$x^5+x^4+x^3$
16	x^4+x^2	38	$x^4+x^3+x^2+1$	60	$x^6+x^5+x^4$
17	x^5+x^3	39	$x^5+x^4+x^3+x$	61	x^6+x^5+x+1
18	x^6+x^4	40	$x^6+x^5+x^4+x^2$	62	x^6+x^2+1
19	x^5+x+1	41	$x^6+x^5+x^3+x+1$	63	x^3+1
20	x^6+x^2+x	42	$x^6+x^4+x^2+1$	64	x^4+x
21	x^3+x^2+x+1	43	x^5+x^3+1	65	x^5+x^2

Table 15. (cont.)

q	x^q	q	x^q	q	x^q
66	x^6+x^3	88	x^6+x^5+x	110	$x^6+x^5+x^2$
67	x^4+x+1	89	x^6+x^2+x+1	111	x^6+x^3+x+1
68	x^5+x^2+x	90	x^3+x^2+1	112	x^4+x^2+1
69	$x^6+x^3+x^2$	91	x^4+x^3+x	113	x^5+x^3+x
70	x^4+x^3+x+1	92	$x^5+x^4+x^2$	114	$x^6+x^4+x^2$
71	$x^5+x^4+x^2+x$	93	$x^6+x^5+x^3$	115	x^5+x^3+x+1
72	$x^6+x^5+x^3+x^2$	94	x^6+x^4+x+1	116	$x^6+x^4+x^2+x$
73	$x^6+x^4+x^3+x+1$	95	x^5+x^2+1	117	$x^5+x^3+x^2+x+1$
74	$x^5+x^4+x^2+1$	96	x^6+x^3+x	118	$x^6+x^4+x^3+x^2+x$
75	$x^6+x^5+x^3+x$	97	x^4+x^2+x+1	119	$x^5+x^4+x^3+x^2+x+1$
76	$x^6+x^4+x^2+x+1$	98	$x^5+x^3+x^2+x$	120	$x^6+x^5+x^4+x^3+x^2+x$
77	$x^5+x^3+x^2+1$	99	$x^6+x^4+x^3+x^2$	121	$x^6+x^5+x^4+x^3+x^2+x+1$
78	$x^6+x^4+x^3+x$	100	$x^5+x^4+x^3+x+1$	122	$x^6+x^5+x^4+x^3+x^2+1$
79	$x^5+x^4+x^2+x+1$	101	$x^6+x^5+x^4+x^2+x$	123	$x^6+x^5+x^4+x^3+1$
80	$x^6+x^5+x^3+x^2+x$	102	$x^6+x^5+x^3+x^2+x+1$	124	$x^6+x^5+x^4+1$
81	$x^6+x^4+x^3+x^2+x+1$	103	$x^6+x^4+x^3+x^2+1$	125	x^6+x^5+1
82	$x^5+x^4+x^3+x^2+1$	104	$x^5+x^4+x^3+1$	126	x^6+1
83	$x^6+x^5+x^4+x^3+x$	105	$x^6+x^5+x^4+x$		
84	$x^6+x^5+x^4+x^2+x+1$	106	$x^6+x^5+x^2+x+1$		
85	$x^6+x^5+x^3+x^2+1$	107	$x^6+x^3+x^2+1$		
86	$x^6+x^4+x^3+1$	108	x^4+x^3+1		
87	x^5+x^4+1	109	x^5+x^4+x		

Split-Search Work Tables

Table 16.

n_1	$x^{12n_1} \bmod (x^7+x+1)$
0	1
1	x^6+x^5
2	$x^6+x^5+x^4+x^3$
3	$x^6+x^5+x^2+x$
4	$x^5+x^4+x^3+x^2+x$
5	$x^6+x^5+x^4$
6	$x^6+x^5+x^3+x$
7	$x^6+x^5+x^4+x^2+x+1$
8	x^6+x^3+x
9	x^4+x^3+1
10	$x^6+x^5+x^4+x^3+x^2+x+1$
11	x^5

Table 17.

$(x^5+x^3+x+1)\cdot(x^6+1)^{n_2} \bmod (x^7+x+1)$	n_2
x^5+x^3+x+1	0
$x^6+x^4+x^2$	1
x^5+x^3+x	2
x^4+x^2+1	3
x^6+x^3+x+1	4
$x^6+x^5+x^2$	5
x^5+x^4+x	6
x^4+x^3+1	7
$x^6+x^3+x^2+1$	8
$x^6+x^5+x^2+x+1$	9
$x^6+x^5+x^4+x$	10
x^4+x	11

The split-search attack is not the only attack, however. Recently, Pollard (1978) presented a randomly driven algorithm to compute the logarithm of an exponentiated primitive root modulo its prime. Pollard's method requires on the order of $p^{1/2}$ computations. Its significant benefit is that, unlike the split-search attack, it does not require the storing and processing of a large quantity of information. Although Pollard did not adapt his algorithm to finite fields other than those formed about primitive roots (integer domains), he stated that such an extension should be easily effected.

Another method has been discovered by Adleman (1979). Adleman's method runs in time proportional to an expression on the order of $\exp(\sqrt{\ln(p)\ \ln(\ln(p))}\ )$. This expression is subexponential and its significance can best be appreciated by the entries in Table 18. Letting $q = 2^a$, Table 18 compares q, $q^{1/2}$ and $\exp(\sqrt{\ln(q)\ \ln(\ln(q))}\ )$ against a.

3. THE DISCRETE LOGARITHM PUBLIC CRYPTOGRAPHIC SYSTEM

3.1 Introduction

We have developed the concept of raising primitive field elements to powers and reducing them modulo prime numbers and primitive polynomials. We have stated that the inverse problem, the index or 'logarithm' problem, i.e., determining the exponent X of α^X given α and α^X, is evidently more difficult than exponentiation. In section 2.7 we hinted that the logarithm problem's difficulty ranges from greater than that of exponentiation to so much greater that, for the present at least, it appears that the logarithm problem is effectively impossible. For those cases in which the logarithm problem is effectively impossible, we can construct a cryptographic system of the 'Public Key' genre.

Diffie and Hellman (1976) proposed a cryptographic system based on integer exponentiation over GF(p), where p is a large prime. We will refer to this system by DLI, an acronym for Discrete Logarithm (the basis of the apparent asymmetric complexity) using Integers.

Table 18. Comparison of Work Required for Three Different Attacks
[x(y) is shorthand for $x \cdot 10^y$]

a	q	$q^{1/2}$	$\exp(\sqrt{\ln(q)\ \ln(\ln(q))}\ )$
20	1.0(6)	1.0(3)	4.2(2)
40	1.1(12)	1.0(6)	1.5(4)
60	1.2(18)	1.1(9)	2.6(5)
80	1.2(24)	1.1(12)	3.0(6)
100	1.3(30)	1.1(15)	2.8(7)
120	1.3(37)	1.2(18)	2.1(8)
140		1.2(21)	1.4(9)
160		1.2(24)	8.4(9)
180		1.2(27)	4.5(10)
200		1.3(30)	2.3(11)
220		1.3(33)	1.1(12)
240		1.3(36)	4.6(12)
260			1.9(13)
280			7.7(13)
300			2.9(14)
320			1.1(15)
340			3.8(15)
360			1.3(16)
380			4.4(16)
400			1.4(17)

Berkovits, Kowalchuk, and Schanning (1979) introduced a variant to the discrete logarithm problem and set forth a system based on exponentiation over $GF(2^n)$ using primitive polynomials. We will refer to this system as DLP, an acronym for Discrete Logarithm using Polynomials.

The procedure underlying both systems is essentially the same. Let there be two parties, A and B, who wish to create a quantity (an integer or polynomial) which is known to them jointly but not to anyone else. Further assume that they possess no mutually private data a priori and that all communication between parties A and B is available to any interested third party. Parties A and B proceed as follows:

PARTY A		PARTY B
	o BOTH PARTIES MUTUALLY AND PUBLICLY AGREE ON ELEMENT α FROM A PUBLICLY AGREED UPON FINITE FIELD	
o PARTY A GENERATES A NUMBER X_A		o PARTY B GENERATES A NUMBER X_B
o PARTY A SAVES X_A AND CONCEALS IT FROM EVERYONE ELSE		o PARTY B SAVES X_B AND CONCEALS IT FROM EVERYONE ELSE
o PARTY A COMPUTES α^{X_A}		o PARTY B COMPUTES α^{X_B}
o PARTY A TRANSMITS α^{X_A} TO PARTY B		o PARTY B TRANSMITS α^{X_B} TO PARTY A
o PARTY A COMPUTES $(\alpha^{X_B})^{X_A}$		o PARTY B COMPUTES $(\alpha^{X_A})^{X_B}$
	o BOTH PARTIES NOW POSSESS $\alpha^{X_A X_B}$	

All that a third party can glean from communications intercept is α, α^{X_A} and α^{X_B}.

3.2 Example of the DLI

An example of the DLI using the prime modulus 127 and the primitive element $\alpha = 3$ is given as follows:

PARTY A		PARTY B
	o BOTH PARTIES AGREE TO USE $\alpha = 3$ AND REDUCE MODULO 127	
o PARTY A CHOOSES (GENERATES IN A RANDOM MANNER) $X_A = 16$		o PARTY B CHOOSES $X_B = 72$
o PARTY A COMPUTES 3^{16} MOD 127 = 71		o PARTY B COMPUTES 3^{72} MOD 127 = 2
o PARTY A TRANSMITS 71 TO PARTY B		o PARTY B TRANSMITS 2 TO PARTY A
o PARTY A COMPUTES 2^{16} modulo 127 = 4		o PARTY B COMPUTES 71^{72} modulo 127 = 4
	o BOTH PARTIES NOW POSSESS A QUANTITY, 4, WHICH IS KNOWN TO THEM ONLY.	

3.3 Example of the DLP

An example of the DLP using the primitive polynomial $x^7 + x + 1$ and the primitive field element $\alpha = x$ is given as follows:

PARTY A		PARTY B
	o BOTH PARTIES AGREE TO USE $\alpha = x$ AND REDUCE MODULO $x^7 + x + 1$	
o PARTY A CHOOSES $X_A = 56$		o PARTY B CHOOSES $X_B = 18$
o PARTY A COMPUTES $x^{56} \bmod(x^7 + x + 1) = x^2 + x + 1$		o PARTY B COMPUTES $x^{18} \bmod(x^7 + x + 1) = x^6 + x^4$
o PARTY A TRANSMITS $x^2 + x + 1$ TO PARTY B		o PARTY B TRANSMITS $x^6 + x^4$ TO PARTY A
o PARTY A COMPUTES $(x^6 + x^4)^{56} \bmod(x^7 + x + 1)$ $= x^5 + x^4 + x^3 + x^2 + x + 1$		o PARTY B COMPUTES $(x^2 + x + 1)^{18} \bmod(x^7 + x + 1)$ $= x^5 + x^4 + x^3 + x^2 + x + 1$
	o BOTH PARTIES NOW POSSESS A QUANTITY, $x^5 + x^4 + x^3 + x^2 + x + 1$, WHICH IS KNOWN TO THEM ONLY.	

3.4 Implementation

To implement either the DLI or the DLP, the implementor must choose a finite field within which to work and a primitive element of the field. The security afforded by the system will be in large part determined by the number of field elements but there are other important considerations specific to the system.

Additional Consideration for the DLI System

From Pohlig and Hellman's work (1978), we note that the type of prime is very important. Pohlig and Hellman's attack works extremely efficaciously against those primes, p, for which p - 1 canonically decomposes into a product of powers of small primes. The fourth Fermat number, 65537, is an excellent example of such a prime. Note that $65536 = 2^{16}$. The best choice for a prime, p, in order to render the Pohlig and Hellman attack impotent, would be a case for which p is a 'safe prime,' i.e., $p = 2p' + 1$ where p' is also a prime.

Additional Consideration for the DLP System

The best choice for a field of 2^n elements is one in which $2^n - 1$ is a prime number. All elements, excepting the additive and multiplicative identities, are primitive for this case. (All irreducible polynomials are also primitive in this case.) Although Pohlig and Hellman's attack (1978) extends from GF(p) to all finite fields, $GF(p^n)$, it produces no advantage over the Split-Search attack for $GF(2^n)$ when $2^n - 1$ is prime.

3.5 Other Security Considerations

3.5.1 Computation Time Attack

If a DLI or DLP system is operated real-time, on-line, it is conceivable that an attack could be mounted by measuring the time required for exponentiation and relating it to the bit density of the exponent. For example, it takes longer to form α^7 from α than to form α^8. If one could determine $\sigma(X_A)$[4] from the time required to compute α^{X_A}, then one need perform only a sequential search of $\binom{n}{\sigma(X_A)}$ cryptovariables where n is the length of the cryptovariable.

The system, therefore, should be implemented so that there will be no way that externally measurable timing information can convey any information regarding the cryptovariable in use.

---

[4]The notation $\sigma()$ is defined in Section 2.6, entitled 'The Art of Exponentiation.'

3.5.2 Active Transparency Attack

The Active Transparancy Attack (ATA) is an attack that can be carried out only if the perpetrator:

(a) is sophisticated.

(b) has access to the telecommunications medium to the extent that he/she is able to interrupt it and insert his/her own traffic.

The attack schema is illustrated in Figure 4 and proceeds as follows:

(a) The perpetrator, Party P, electronically inserts himself between Party A and Party B.

(b) Party A sends α^{X_A} to Party B. Party B never receives α^{X_A}. Party P, however, does receive α^{X_A}.

(c) Party P sends Party A $\alpha^{X_{P1}}$.

(d) Party P sends Party B $\alpha^{X_{P2}}$.

(e) Party P computes $(\alpha^{X_A})^{X_{P1}}$.

(f) Party A computes $(\alpha^{X_{P1}})^{X_A}$.

(g) Party B sends Party A α^{X_B}.

(h) Party A never receives α^{X_B}. Party P, however, does receive α^{X_B}.

(i) Party B computes $(\alpha^{X_{P2}})^{X_B}$.

(j) Party B computes $(\alpha^{X_B})^{X_{P2}}$.

As a result, Parties A and P hold $\alpha^{X_A X_{P1}}$ in common and Parties B and P hold $\alpha^{X_B X_{P2}}$ in common. If Party A (or B) sends traffic to Party B (or A), Party P will decipher the traffic, exploit it, and reencipher the traffic using the other commonly held cryptovariable and send it on to the other legitimate party.

3.5.3 ATA Countermeasure

A countermeasure to the ATA has been suggested by McRae (private communication). McRae's idea requires Party A or B to send an unused part of $\alpha^{X_A X_B}$ to Party B or A by another channel or to use another authentication scheme such as voice recognition.

How this procedure would help defeat the ATA becomes obvious when we consider

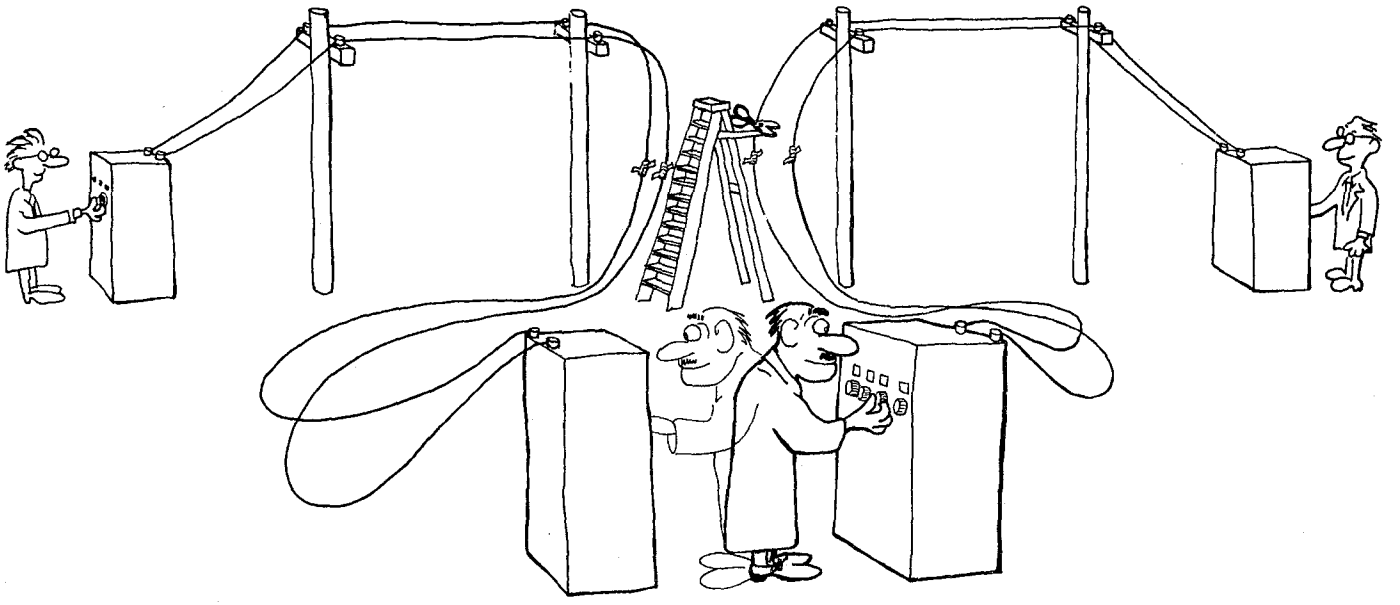

Figure 4. The Active Transparency Attack.

that, under the ATA, Parties A and B share no secret quantity; rather, all traffic is unencrypted and then reencrypted by Party P.

As an example of the ATA countermeasure, consider that:

(a) Party P has transparently interposed himself between Parties A and B.

(b) Parties A and P and Parties B and P agree to use the DLP with $\alpha = x$ and reduce modulo $x^7 + x + 1$.

(c) Party A chooses $X_A = 20$ and computes $x^{20} \bmod(x^7 + x + 1) = x^6 + x^2 + x$.

(d) Party P chooses $X_{P1} = 43$ and computes $x^{43} \bmod(x^7 + x + 1) = x^5 + x^3 + 1$.

(e) Party A sends $x^6 + x^2 + x$ to Party P.

(f) Party P sends $x^5 + x^3 + 1$ to Party A.

(g) Party A computes $(x^5 + x^3 + 1)^{20} \bmod(x^7 + x + 1) = x^5 + x^3 + x^2 + x$.

(h) Party P computes $(x^6 + x^2 + x)^{43} \bmod (x^7 + x + 1) = x^5 + x^3 + x^2 + x$.

(i) Party B chooses $X_B = 80$ and computes $x^{80} \bmod(x^7 + x + 1) = x^6 + x^5 + x^3 + x^2 + x$.

(j) Party P chooses $X_{P2} = 53$ and computes $x^{53} \bmod(x^7 + x + 1) = x^6 + x^4 + 1$.

(k) Party P sends $x^6 + x^4 + 1$ to Party B.

(l) Party B sends $x^6 + x^5 + x^3 + x^2 + x$ to Party P.

(m) Party B computes $(x^6 + x^4 + 1)^{80} \bmod(x^7 + x + 1) = x^6 + x^5 + x^4 + x^3 + x^2$.

(n) Party P computes $(x^6 + x^5 + x^3 + x^2 + x)^{53} \bmod(x^7 + x + 1) = x^6 + x^5 + x^4 + x^3 + x^2$.

If it is standard procedure for Parties A and B to authenticate, by voice or other channel, by exchanging (publicly) the coefficients of x^2, x, and 1 (remember that Parties A and B do <u>not</u> use these three bits of their common secret quantity as key for any secret process), then A and B will detect a mismatch as A will hold 1, 1, 0, and B will hold 1, 0, 0.

3.6 Architectural Tools

We have already shown that when implementing the DLP using polynomials over $GF(2^m)$, it is best to choose m such that $2^m - 1$ is a large prime. (A prime of this form is termed a Mersenne prime.) Two things are needed. First, we need a list of Mersenne primes and, second, we need to know a primitive polynomial of the Mersenne prime degree.

The first twenty-seven Mersenne primes are known. The following table, compiled from Zierler (1969) and Noll and Nickel (1980), presents the primes and:

(a) if primitive trinomials exist, the trinomials themselves.

(b) the notation 'NONE EXIST' if no primitive trinomials exist of the Mersenne prime order.

(c) a blank if the primitive trinomial question is unanswered.

p, for which $2^p - 1$ is prime	primitive trinomials[5]
2	$x^2 + x + 1$
3	$x^3 + x + 1$
5	$x^5 + x^2 + 1$
7	$x^7 + x + 1$
	$x^7 + x^3 + 1$
13	NONE EXIST[6]
17	$x^{17} + x^3 + 1$
	$x^{17} + x^5 + 1$
	$x^{17} + x^6 + 1$
19	NONE EXIST
31	$x^{31} + x^3 + 1$
	$x^{31} + x^6 + 1$
	$x^{31} + x^7 + 1$
	$x^{31} + x^{13} + 1$
61	NONE EXIST

[5] If $x^p + x^a + 1$ is a primitive trinomial then so is $x^p + x^{p-a} + 1$. Only the former is listed.

[6] There do, however, exist primitive polynomials of this and all other degrees.

p, for which $2^p - 1$ is prime	primitive trinomials
89	$x^{89} + x^{38} + 1$
107	NONE EXIST
127	$x^{127} + x + 1$
	$x^{127} + x^{7} + 1$
	$x^{127} + x^{15} + 1$
	$x^{127} + x^{30} + 1$
	$x^{127} + x^{63} + 1$
521	$x^{521} + x^{32} + 1$
	$x^{521} + x^{48} + 1$
	$x^{521} + x^{158} + 1$
	$x^{521} + x^{168} + 1$
607	$x^{607} + x^{105} + 1$
	$x^{607} + x^{147} + 1$
	$x^{607} + x^{273} + 1$
1279	$x^{1279} + x^{216} + 1$
	$x^{1279} + x^{418} + 1$
2203	NONE EXIST
2281	$x^{2281} + x^{715} + 1$
	$x^{2281} + x^{915} + 1$
	$x^{2281} + x^{1029} + 1$
3217	$x^{3217} + x^{67} + 1$
	$x^{3217} + x^{576} + 1$
4253	NONE EXIST
4423	$x^{4423} + x^{271} + 1$
	$x^{4423} + x^{369} + 1$
	$x^{4423} + x^{370} + 1$
	$x^{4423} + x^{649} + 1$

p, for which $2^p - 1$ is prime	primitive trinomials
	$x^{4423} + x^{1393} + 1$
	$x^{4423} + x^{1419} + 1$
	$x^{4423} + x^{2098} + 1$
9689	$x^{9689} + x^{84} + 1$
	$x^{9689} + x^{471} + 1$
	$x^{9689} + x^{1836} + 1$
	$x^{9689} + x^{2444} + 1$
	$x^{9689} + x^{4187} + 1$
9941	NONE EXIST
11213	NONE EXIST
19937	
21701	
23209	
44497	

4. SUMMARY AND CONCLUSIONS

We have described a very important subset of present-day Public Cryptographic methods, specifically those systems built upon the apparent asymmetric complexity between finite field exponentiation and the taking of discrete 'logarithms' in a finite field. We have presented the requisite mathematics and have attempted to be very liberal in our use of examples to highlight the mathematical preliminaries and the cryptographic systems themselves. The most important issue, the cryptographic strength, the systems' resistance to cryptanalysis, has been addressed and finally we have provided advice regarding the architecture of system implementation and specific architectural tools that may be used.

5. ACKNOWLEDGMENTS

The author thanks Dr. Peter M. McManamon for his encouragement and support in working in this area, Dr. William Hartman for many helpful discussions, and Mr. Rex Powell for graphics services.

6. REFERENCES

Adleman, L. (1979), A subexponential algorithm for the discrete logarithm problem with applications to cryptography, 20th Annual Symposium on Foundations of Computer Science, October.

Albert, A. A. (1956), Fundamental Concepts of Higher Algebra, Phoenix Books (The University of Chicago Press).

Beiler, A. (1966), Recreations in the Theory of Numbers, (Dover Press).

Berkovits, S., J. Kowalchuk, and B. Schanning (1979), Implementing public key scheme, IEEE Communications Magazine 17, No. 3, May.

Bouniakowsky, V. (1870), Bulletin of the Academy of Sciences, St. Petersburg 14.

Dean, R. A. (1966), Elements of Modern Algebra, (Wiley & Sons).

Diffie, W. and M. Hellman (1976), New directions in cryptography, IEEE Trans. Information Theory IT-22, No. 6, November.

Golomb, S. (1967), Shift Register Sequences, (Holden-Day).

Hershey, J. E. (1980), Implementation of MITRE public key cryptographic system, Electron. Letters 16, No. 24, November.

Kautz, W. H. (editor) (1965), Linear Sequential Switching Circuits, (Holden-Day).

Knuth, D. E. (1969), The Art of Computer Programming, Vol. 2 (Seminumerical Algorithms), (Addison-Wesley).

Knuth, D. E. (1973), The Art of Computer Programming, Vol. 3 (Sorting and Searching), (Addison-Wesley).

LeVeque, W. J. (1956), Topics in Number Theory, Vol. I, (Addison-Wesley).

Noll, C., and L. Nickel (1980), The 25th and 26th Mersenne Primes, Mathematics of Computation 35, No. 152, October.

Perlis, S. (1952), Theory of Matrices, (Addison-Wesley).

Pohlig, S., and M. Hellman, (1978), An improved algorithm for computing logarithms over GF(p) and its cryptographic significance, IEEE Trans. Information Theory IT-24, No. 1, January.

Pollard, J. M. (1978), Monte Carlo methods for index computation (mod p), Mathematics of Computation 32, No. 143, July.

Van derWaerden, B. L. (1953), Modern Algebra, (Frederick Ungar Publishing Co., NY).

Zierler, N. (1969), Primitive Trinomials whose Degree is a Mersenne Exponent, Information and Control 15.

FORM **NTIA-29**
(4-80)

U.S. DEPARTMENT OF COMMERCE
NAT'L. TELECOMMUNICATIONS AND INFORMATION ADMINISTRATION

BIBLIOGRAPHIC DATA SHEET

	1. PUBLICATION NO.	2. Gov't Accession No.	3. Recipient's Accession No.
	NTIA Report 81-81		

4. TITLE AND SUBTITLE
The Discrete Logarithm Public Cryptographic System

5. Publication Date
September 1981

6. Performing Organization Code
NTIA/ITS-4

7. AUTHOR(S)
John E. Hershey

9. Project/Task/Work Unit No.

8. PERFORMING ORGANIZATION NAME AND ADDRESS
U.S. Department of Commerce
NTIA/ITS-4
325 Broadway
Boulder, Colorado 80303

10. Contract/Grant No.

11. Sponsoring Organization Name and Address
NTIA/ITS-4
325 Broadway
Boulder, Colorado 80303

12. Type of Report and Period Covered

13.

14. SUPPLEMENTARY NOTES

15. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)*

The report is a study and primer of the discrete logarithm public key cryptographic system. Implementation and strengths and weaknesses are discussed.

16. Key Words *(Alphabetical order, separated by semicolons)*

Cryptography; Diffie-Hellman system; finite field logarithms; MITRE system; public key cryptography

17. AVAILABILITY STATEMENT

☒ UNLIMITED.

☐ FOR OFFICIAL DISTRIBUTION.

18. Security Class. *(This report)*
Unclassified

20. Number of pages
48

19. Security Class. *(This page)*
Unclassified

21. Price:

☆ U.S. GOVERNMENT PRINTING OFFICE:1981—781-822 / 561