SECTION 4

Cryptanalysis

CHAPTER 10

Cryptanalysis
A *Survey of Recent Results*

E. F. BRICKELL
Sandia National Laboratories
Albuquerque, New Mexico 87185

A. M. ODLYZKO
A T & T Bell Laboratories
Murray Hill, New Jersey 07974

***Abstract*-In spite of the progress in computational complexity, it is still true that cryptosystems are tested by subjecting them to cryptanalytic attacks by experts. Most of the cryptosystems that have been publicly proposed in the last decade have been broken. This chapter outlines a selection of the attacks that have been used and explains some of the basic tools available to the cryptanalyst. Attacks on knapsack cryptosystems, congruential generators, and a variety of two-key secrecy and signature schemes are discussed. There is also a brief discussion of the status of the security of cryptosystems for which there are no known feasible attacks, such as the Rive&Shamir-Adleman (RSA), diirete exponentiation, and Data Encryption Standard (DES) cryptosystems.**

1 INTRODUCTION

The last decade has seen explosive growth in unclassified research in all aspects of cryptology, and cryptanalysis has been one of the most active areas. Many cryptosysterns that had been thought to be secure have been broken, and a large collection of mathematical tools useful in cryptanalysis has been developed. The purpose of this survey is to present some of the recent attacks in a way that explains and systemizes the cryptanalytic techniques that are used, with the hope that they will be useful in assessing the security of other cryptosystems.

Most of the discussion in this chapter is devoted to public key systems. This reflects the general developments in cryptography over the last decade. At the beginning of the 1970s only classic (single-key) cryptography was known, but very little unclassified research was being done on it. The reasons for this lack of interest were manifold.

There did *not* seem to be much need for commercial encryption. The vast body of classified work in cryptography discouraged researchers who naturally like to discover new results. Finally, perhaps the most important factor was that despite the development of the beautiful Shannon theory of secrecy systems, and the use of some tools from abstract algebra, generally speaking cryptography appeared to consist of a large bag of tricks, without a coherent mathematical framework.

The situation changed drastically in the 1970s. First, with growth in communications and the proliferation of computers, the need for cryptographic protection became widely recognized. Second, the invention of public key cryptography by Diffie and Hellman appeared to provide an answer to the commercial need for security that avoided some of the disadvantages of classical cryptography, such as the difficulty of key management. Furthermore, this development galvanized the research community **because** it appeared to open up a brand new field, and it presented the exciting promise of using new tools from the rapidly developing field of computational complexity to develop systems with simple mathematical descriptions. The security of these systems would depend on the intractability of well-known problems, and hopefully would eventually lead to proofs of unbreakability of such systems.

Ironically, the promise of provable security through reduction to well-known mathematical problems has not only not been fulfilled, but instead, the fact that attacks on the new cryptosystems could be formulated as mathematically attractive problems, and that various tools from computational complexity, number theory, and algebra could be brought to bear on them, has resulted in the breaking of many systems. The old one-time pad remains the only system that is known to be unconditionally secure.

The ideal proof of security for a public key cryptosystem would be to show that any attack that has a nonnegligible probability of breaking the system requires an infeasible amount of computation. While no public key system has been shown to satisfy this strong definition of security, the situation is not completely bleak. Many systems have been developed whose security has been proved to be equivalent to the intractability of a few important problems, such as factoring integers, that are almost universally regarded as very hard. (Many of the systems that have been broken were derived from these presumably secure ones by weakening them to obtain greater speed.) Furthermore, the extensive work of the last decade, both in cryptography itself and in general computational complexity, has given cryptologists a much better understanding of what makes a system insecure. The aim of this chapter is to distill some of the lessons of this research.

Before outlining the contents of this chapter, we have to explain what we mean by saying that a cryptographic system is insecure. One can define a fairly precise notion in terms of a polynomial fraction of instances of the system being decipherable in polynomial time. Such an approach is unsatisfactory for two reasons, however. One is that in practice one has to build systems of fairly limited size, and so one cannot assume that asymptotic properties apply. A more serious reason is that for many cryptanalytic attacks, no rigorous proofs of effectiveness exist. Instead one relies on heuristics and experimental evidence; for example, one shows that a reduced-size version of a proposed cryptosystem can be broken relatively fast on a small general-purpose computer, and then one argues that since the effort involved in the attack does not increase too fast with the size of the problem, even the full-size cryptosystem is insecure from a determined attacker. This approach is occasionally used also in other areas of computational complexity (e.g., factoring polynomials or integers), where the best practical algorithms rely on unproved assumptions. Use of such approaches in cryptography is very easy to justify. Because cryptosystems often protect very sensitive information and once adapted, are difficult to change, it is important that they be above suspicion. We will see below, for example, that attacks on some of knapsack cryptosystems depend on being able to find very short nonzero vectors in lattices. In general, it is not known just how difficult a task it is to find such vectors, and the known polynomial time algorithms are not guaranteed to find such vectors. On the other hand, these algorithms usually work much better than they are guaranteed to, and moreover, there has been a lot of progress recently on obtaining improved algorithms. Therefore it seems prudent in assessing the security of the knapsack cryptosystems to assume that one can find even the shortest nonzero vector in a lattice relatively fast.

The remainder of this chapter is organized as follows: Sections 2 and 3 discuss the cryptanalysis of knapsack cryptosystems. Section 4 contains the cryptanalysis of the Ong, Schnorr, and Shamir (OSS) signature schemes, and Section 5 that of the Okamoto-Shiraishi scheme. Section 6 briefly mentions several two-key cryptosystems that have been broken. Sections 7-9 describe what is known about the security of the Rivest-Shamir-Adleman (RSA) algorithms, discrete exponentiation, and the McEliece cryptosystem. The next two sections deal with the cryptanalysis of some single key systems, with Section 10 covering the remarkable success in breaking congruential generators, and Section 11 discussing the remarkable lack of success in breaking DES. Section 12 briefly discusses the successful cryptanalysis of the fast data encipherment algorithm (FEAL). Finally, Section 13 contains some miscellaneous comments.

2 KNAPSACK CRYPTOSYSTEMS

Knapsack cryptosystems are based on the knapsack (or more precisely the subset sum) problem, that is, given a set of integers (or weights) $a_1, \ldots, a_n$ and a specified sum s, find a subset of $\{a_1, \ldots, a_n\}$ that sums to exactly s, or equivalently find a 0–1 vector $(x_1, \ldots, x_n)$ such that $\Sigma_{i=1}^n x_i a_i = s$. We will sometimes refer to the set of weights as a knapsack. Merkle and Hellman [105] discovered a way to use the knapsack problem as the basis for a two-key cryptosystem. Although the knapsack problem is NP-hard [58], there are knapsacks for which the problem is easy. An example of an easy knapsack which was used in [105] is a superincreasing sequence, that is, a sequence of positive integers $b_1, \ldots, b_n$ such that $b_j > \Sigma_{i<j} b_i$, for $1 < j \leq n$.

The basic technique for using the knapsack problem as a two-key cryptosystem is straightforward.

- Public key: Positive integers a,, . . . , a,.
- Private key: A method for transforming $a_1, \ldots, a_n$ into an easy knapsack.
- Message space: n-dimensional 0–1 vectors $(x_1, \ldots, x_n)$.
- Encryption: $s = \Sigma_{i=1}^n x_i a_i$.
- Decryption: Solve the knapsack problem with weights $a_1, \ldots, a_n$ and sum s.

Merkle and Hellman used a superincreasing sequence as an easy knapsack and disguised it with one or more modular multiplications. Specifically, an easy knapsack

$b_1, \ldots, b_n$ can be disguised with a modular multiplication by selecting $M > \Sigma_{i=1}^n b_i$ and W with $(W, M) = 1$, and computing

$$a_i \equiv b_i W \pmod{M} \tag{2.1}$$

Any solution $(x_1, \ldots, x_n)$ to the knapsack problem $\Sigma_{i=1}^n x_i a_i = s$ is also a solution to the knapsack problem $\Sigma_{i=1}^n x_i b_i = s'$ where $s' \equiv sW^{-1} \pmod{M}$ and $0 \leq s' < M$. Merkle and Hellman [105] further observed that the disguising operation could be *iterated* many times. For instance, given the above knapsack $a_1, \ldots, a_n$, a new knapsack $c_1, \ldots, c_n$ could be formed by choosing a new modulus $M_2 > \Sigma_{i=1}^n a_i$ and multiplier W_2 with $(W_2, M_2) = 1$ and defining $c_i \equiv a_i W_2 \bmod M_2$. The knapsack $c_1, \ldots, c_n$ is called a *double-iterated knapsack.*

The designer of the system can further complicate matters by permuting the weights before publishing them. For clarity of exposition, we will assume that the weights are not permuted, but we will discuss the permutation when it is relevant.

One reason to be suspicious about the security of knapsack cryptosystems is that they are basically linear. Specifically $\Sigma_{i=1}^n x_i a_i + \Sigma_{i=1}^n y_i a_i = \Sigma_{i=1}^n (x_i + y_i) a_i$. In fact, if (as we may assume) not all the a_i are even, then by looking at the least significant bit of the ciphertext s we obtain a bit of information about the plaintext, although this usually does not yield even a single bit of the plaintext. Although there is no attack on the cryptosystem based just on this linearity, it should raise questions about its security because linearity in cryptosystems is known to be dangerous. Another cause for suspicion is due to a result of Brassard [18]. Essentially it says that if the problem of breaking a cryptosystem is **NP**-hard, then **NP** = **CoNP**. When the Merkle–Hellman knapsack cryptosystem was proposed, the only attack known was to use an algorithm which would solve any knapsack problem. If one believes that **NP** ≠ **CoNP**, then it seems likely that there is an attack on the Merkle–Hellman knapsack cryptosystem that runs faster than algorithms that solve the general knapsack problem. This suspicion does not apply to RSA, since factoring integers is not believed to be **NP**-hard.

These suspicions were extended by various authors. Herlestam [71] observed by using simulations that often a single bit of the message could be easily recovered. Shamir [150] showed that Merkle–Hellman knapsacks in which the modulus M has close to n bits can be broken easily, and [149] that compact knapsacks (i.e., general knapsacks with few weights c_i and with coefficients x_i that are allowed to vary over a wider range than just the set $\{0, 1\}$) ought to be avoided. Amirazizi, Karnin, and Reyneri [6] also showed that compact knapsacks are insecure, but with an even more powerful argument than that of [149], since they were able to use the theorem of H. W. Lenstra [97] that integer programming in a fixed number of variables is solvable in polynomial time. (This key result [97] was proved at the end of 1980 and became widely known right away, although it was not published until much later.) Shamir and Zippel [151] showed using continued fractions that if the modulus M was known, a cryptanalyst could break the single iterated system. Ingemarsson [73] developed a method of successive reduction modulo suitably chosen integers which seemed to apply to a wide class of knapsacks. However, none of these attacks could convincingly be shown to apply to the Merkle–Hellman system.

Eier and Lagger [51] and independently Desmedt, Vandewalle, and Govaerts [46] made a key observation that led eventually to the complete demise of these knapsack systems. From Eq. (2.1), there exist integers $k_1, \ldots, k_n$ such that

$$a_iU - k_iM = b_i \tag{2.2}$$

where $U \equiv W^{-1} \bmod M$. Therefore

$$\frac{U}{M} - \frac{k_i}{a_i} = \frac{b_i}{a_iM} \tag{2.3}$$

and so all of the k_i/a_i are close to U/M. Furthermore, as was apparently realized by the authors of [51] and [46], the actual values of U and M are not important, since if one finds any pair of integers u and m with u/m U/M small, one can use u and m to decrypt the knapsack. For an arbitrary collection of integers a_i it is highly unlikely that there would exist k_i such that all of the k_i/a_i would be close together. This seemed to provide a way to attack the Merkle-Hellman system.

Shamir [146] completed the cryptanalysis of the single iterated Merkle-Hellman system by making two more observations. Since the a_i's are superincreasing, $a_i < M2^{i-n}$. Hence, for small i, the k_i/a_i are extremely close together; from Eq. (2.2) we see that

$$|b_ik_1 - b_1k_i| \leq M2^{i-n} \tag{2.4}$$

Only a few (three to four) of these inequalities uniquely determine the k_i's, and once the k_i's are found, it is easy to break the system. The system (Eq. (2.4)) is an instance of integer programming with a small number of variables. Therefore the Lenstra [97] integer linear programming algorithm can find the k_i's fast. For this attack, it is necessary for the cryptanalyst to know which of the public weights correspond to the smallest elements in the superincreasing sequence. If the knapsack was permuted before it was published, he would not know this. Since he only needs to know the three or four smallest elements, however, he can find them in polynomial time ($0(n^3)$ or $0(n^4)$) by trying all possibilities.

The Shamir attack sketched above was universally accepted as valid when it was announced, although nobody up to that time had implemented the Lenstra integer programming algorithm. (In fact, as will be explained below, for the standard version of the Merkle-Hellman system, in which M has about $2n$ bits, one can use continued fractions to find the k_i.) Furthermore, the Shamir attack did not seem to generalize to other knapsack systems. These problems were soon overcome, though, because Adleman [3] found that the Lovasz lattice basis reduction algorithm [93] could be used instead of the Lenstra integer programming algorithm, and this enabled him to break the Graham-Shamir knapsack cryptosystem (see [151] for a definition). The introduction of this new tool, the Lovasz algorithm, was the main key to most of the major breakthroughs that were achieved in analyzing knapsacks. The Lovasz algorithm and more efficient ones that were derived later by Radziszowski and Kreher [134] and by Schnorr ([140,141]) are now among the basic tools of constructive diophantine approximation, and will be discussed after we introduce some definitions.

2.1 Diophantine Approximation

Simultaneous diophantine approximation is the study of approximating a vector of reals $(\theta_1, \ldots, \theta_n)$ by a vector of rationals $(\frac{p_1}{p}, \ldots, \frac{p_n}{p})$ all having the same denominator. An approximation $(\frac{p_1}{p}, \ldots, \frac{p_n}{p})$ to a vector of rationals $(\frac{q_1}{q}, \ldots, \frac{q_n}{q})$ is said to be an unusually good simultaneous diophantine approximation (UGSDA), if $|p\frac{q_i}{q} - p_i| \leq q^{-\delta}$ for some

$\delta > \frac{1}{n}$. Lagarias [86] has justified this definition by showing that unusually good simultaneous diophantine approximations are indeed unusual.

For breaking knapsack-type cryptosystems, we are interested in the algorithmic

known to exist. For $n = 1$, continued fractions can be used to find UGSDA. The set of convergents to the continued fraction expansion of $\frac{q_1}{q}$ contains every rational $\frac{r_1}{r}$ such that $r < q$ and $\left|r\frac{q_1}{q} - r_1\right| \leq \frac{1}{r}$. Thus if $\frac{p_1}{p}$ is an UGSDA to $\frac{q_1}{q}$ then $\frac{p_1}{p}$ will be a convergent.

More surprisingly, continued fractions can also be used to find UGSDA for $n = 2$. To see this let $\left(\frac{q_1}{q}, \frac{q_2}{q}\right)$ be a pair of rationals that have an UGSDA $\left(\frac{p_1}{p}, \frac{p_2}{p}\right)$. Let

$$c_i = q_i p - q p_i$$

Then

$$|c_i| < q^{\frac{1}{2}}$$

Taking these equations mod q, we obtain

$$c_i \equiv q_i p \bmod q$$

Let us assume for now that the greatest common divisor GDC$(q_2, q) = 1$. Then

$$\frac{c_1}{c_2} = \frac{q_1}{q_2} \text{ mod q}$$

Let $x \equiv \frac{q_1}{q_2} \bmod q$. Then $c_2 x \equiv c_1 \bmod q$, and there exists a y such that

$$c_2 \frac{x}{q} - y = \frac{c_1}{q}$$

and

$$\left|\frac{c_1}{q}\right| < |c_2|^{-1}$$

Therefore we can find $\frac{y}{c_2}$ as a convergent in the continued fraction expansion of $\frac{x}{q}$. Using c_2, we can find $p \equiv q_2^{-1} c_2 \bmod q$ and then p_1 and p_2 are easily determined. If the GCD $(q_2, q) = d \neq 1$, then one can replace q and q_2 by $\frac{q}{d}$ and $\frac{q_2}{d}$ and proceed as above. This provides an attack on the single iterated Merkle-Hellman cryptosystem for certain parameters. In particular, if $M < 2^{2n-8}$and if $b_1 > M/2$, then using Eq. (2.4) we see that $\left(\frac{k_2}{k_1}, \frac{k_3}{k_1}\right)$ is an UGSDA to $\left(\frac{b_2}{b_1}, \frac{b_3}{b_1}\right)$. The knapsack cryptosystem of Henry [70] can be broken by using continued fractions.

Finding UGSDA is related to finding short vectors in a lattice. Given a set of n independent vectors in R^n,$b_1, \ldots, b_n$, a lattice L is the set of points

$$L = \left\{ \sum_{i=1}^{n} z_i \, \mathbf{b}_i : z_i \in \mathbb{Z} \right\}$$

The vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$ are said to be a basis for L. Consider the lattice L generated by the row vectors $\mathbf{b}_0, \ldots, \mathbf{b}_n$ of the matrix

$$\begin{pmatrix} \lambda & p_1 & p_2 & \cdots & p_{n-1} & p_n \\ 0 & -p & 0 & \cdots & 0 & 0 \\ \vdots \\ 0 & 0 & 0 & \cdots & -p & 0 \\ 0 & 0 & 0 & \cdots & 0 & -p \end{pmatrix}$$

where A is a real number between 0 and 1. There is an obvious relationship between short vectors in this lattice and UGSDA to $(\frac{p_1}{p}, \ldots, \frac{p_n}{p})$ since a vector $\mathbf{v} = \Sigma_{i=0}^{n} q_i\mathbf{b}_i$ in L has length $\|\mathbf{v}\| = \sqrt{\Sigma_{i=1}^{n}(q_0 p_i - q_i p)^2 + \lambda^2 q_0^2}$. ($\lambda$ should be chosen small enough so that λ_{q_0} is not the largest term in this sum.)

Although the problem of finding the shortest vector in a lattice is not known to be NP-hard, there is no known polynomial time algorithm for solving it. There are, however, polynomial time algorithms for finding relatively short vectors in a lattice. The first such algorithm was due to Lovasz [93]. For a lattice with a basis in which all coefficients in the basis are integers with absolute value < B, the algorithm is guaranteed to terminate in O(n^6(log B)^3) bit operations and produce a vector v such that $\|\mathbf{v}\|^2 \leq 2^n \|\mathbf{u}\|^2$ where $\mathbf{u}$ is the shortest nonzero vector in the lattice. In practice, mod-

$O(n(\log B)^3)$

steps, and produce vectors that are much closer to the length of the shortest vector in the lattice. In particular, on all of a small set of test cases which came from knapsack cryptosystems, an implementation of the Lovasz algorithm by Brickell [21] found vectors that could be used to break the cryptosystem even though the vectors needed were only about l/n times the length of the original basis vectors in the lattice. There are also other lattice basis reduction algorithms due to Schnorr [141] that produce vectors that are guaranteed to be closer in length to the shortest vector in the lattice, but these algorithms are slower.

2.2 Multiple-Iterated Knapsacks

An I-iterated knapsack cryptosystem is one in which I modular multiplications are used to disguise an easy knapsack. For an I-iterated knapsack, there are I independent UGSDA. These UGSDA were studied by Brickell, Lagarias, and Odlyzko [25] and more extensively by Lagarias [86,87] and were later used to break the multiple iterated knapsack by Brickell [21].

These UGSDA can be used to break all of the knapsack cryptosystems [8,20,47,50,115,123,143,144,145,161] that have been proposed that rely on modular multiplications as a disguising technique. See the surveys by Brickell [23] and Desmedt [44] for more details.

2.3 Low-Density Attacks

The density of a knapsack $a_1, \ldots, a_n$ in which $A = \max\{a_1, \ldots, a_n\}$ is defined to be $\frac{n}{\log_2(A)}$. There are algorithms due to Brickell [19] and to Lagarias and Odlyzko [88] for solving knapsacks of low density. The Lagarias-Odlyzko [88] algorithm consists of looking for short vectors in the lattice L generated by the row vectors in the matrix

$$\begin{pmatrix} 1\,0 & \ldots & 0\ a_1 \\ 0\,1 & \ldots & 0\ a_2 \\ \cdot & & \\ \cdot & & \\ \cdot & & \\ 0\,0 & \ldots & 1\ a_n \\ 0\,0 & \ldots & 0\ s \end{pmatrix}$$

where s is the sum for the knapsack problem. In [88], the algorithm is analyzed with the Lovasz basis reduction algorithm [93] being used to find the short vectors in L. *The* **polynomial time algorithm will solve almost all knapsack problems of density** $< \frac{2}{n}$. (Frieze [54] has obtained a simpler proof of this result.) In practice, the algorithm is successful on knapsacks of much higher density, but the densities for which the algorithm succeeds does appear to go to 0 as n increases. Using more efficient lattice basis reduction algorithms [134,140,141] would increase the critical density below which this attack succeeds.

3 GENERALIZED KNAPSACK CRYPTOSYSTEMS

In this section, we will examine several cryptosystems that have been proposed which use ideas similar to those used in the knapsack cryptosystems.

3.1 Lu-Lee Systems

The Lu-Lee Cryptosystem [98]:

- **Public key:** c_1, c_2, r, M_1, M_2 **all positive integers.**
- **Messages: Integers** m_1, m_2 **such that** $0 < m_1 < M_1$, $0 < m_2 < M_2$.
- Encryption:

$$E(m_1, m_2) = c_1 m_1 + c_2 m_2 \bmod r \tag{3.1.1}$$

- Decryption: The parameters were chosen so that the encryption function is one-to-one on the message space. The knowledge of the private key allows easy decryption.

This cryptosystem was broken by Adleman and Rivest [4], Goethals and Couvreur [60], and Kochanski [81]. Adiga and Shankar [2] suggested a modification of this scheme.

The modified Lu-Lee cryptosystem [2]:

- **Public key:** c_1, c_2, r, M **all positive integers.**
- **Messages: Positive integers** $m < M$.
- **Encryption: Pick** $m_1 < M_1$, $m_2 < M_2$, **and compute**

$$E(m) = m + c_1 m_1 + c_2 m_2 \bmod r \tag{3.1.2}$$

- Decryption: Same remarks as above apply.

For both of these systems, cryptanalysis by solving Eqs. (3.1.1) or (3.1.2) by integer linear programming is immediate. Kannan's integer linear programming algorithm [78] runs in $O(n^{9n} \log r)$ in the worst case on problems with n variables and integer coefficients bounded by r. Since $n \leq 4$ for Eqs. (3.1.1) and (3.1.2), Kannan's algorithm is a viable threat to these systems.

3.2 Niederreiter Cryptosystem

Niederreiter [114] proposed a knapsack-type cryptosystem using algebraic coding theory.

- Private key: H, an $(n - k)$ by n parity check matrix of a f-error correcting linear (n, k) code, C, over GF(q) with an efficient decoding algorithm. P, an n X n permutation matrix. M, a nonsingular $(n - k)$ X $(n - k)$ matrix.
- Public key: $K = MHP$ and t.
- Messages: n dimensional vectors **y** over GF(q) with weight $\leq t$.
- Encryption: $\mathbf{z} = K\mathbf{y}^T$.
- Decryption: Since $\mathbf{z} = K\mathbf{y}^T = MHP\mathbf{y}^T$, $M^{-1}\mathbf{z} = HP\mathbf{y}^T = H(\mathbf{y}P^T)^T$. Use the decoding algorithm for C to find $\mathbf{y}P^T$ and thus **y**.

This cryptosystem is said to be of knapsack type because the encryption can be viewed as picking t columns from the matrix K and forming a weighted sum of these t column vectors.

We will mention three cryptanalytic attacks on this system. In the first attack, for a ciphertest, z, we pick a submatrix J of K consisting of $(n - k)$ columns of K. We then compute $\mathbf{y}' = J^{-1}\mathbf{z}$. If all of the t columns that were added to form **z** are in J, then $\mathbf{y}'$ will be the encrypted message, that is, y' will satisfy $Ky' = z$ and have at most t nonzero entries. The probability of this occurrence is $p = \binom{n-k}{t} / \binom{n}{t}$. Thus the expected number of times we must repeat this procedure before we are successful is $\frac{1}{p}$. There are two examples mentioned in [114]. For the first $n = 104$, $k = 24$, $t = 15$, and so $\frac{1}{p} = 72$. For the second example, $n = 30$, $k = 12$, $t = 9$, and $\frac{1}{p} = 295$.

Another attack on this cryptosystem is based on a deterministic linear algebra procedure. It is easy to find some vector **w** such that $K\mathbf{w} = \mathbf{z}$. Once **w** is found, we must have $\mathbf{w} = \mathbf{y} + \mathbf{c}$, for some codeword **c** in C. We can write C as the direct sum of two subspaces C_1 and C_2, with C_1 of dimension $[k/2]$, and list all the codewords of C_1 and C_2 (approximately $q^{k/2}$ in each case). Then, for each $\mathbf{c}_1$ in C_1, we only need to check whether $\mathbf{w} - \mathbf{y} - \mathbf{c}$ is in C_2. In both of the examples presented in [114], this procedure would be very fast.

There is another attack based on the low-density algorithm of [88] that can be used if GF(q) is a prime field, that is, q is a prime. Let $\mathbf{v}_1^T, \ldots, \mathbf{v}_n^T$ be the n column vectors of K. Let $v_i = (v_{i1}, v_{i2}, \ldots, v_{i,n-k})$. Let r be an integer. Let L be the lattice generated by the row vectors in the matrix

$$Q = \begin{pmatrix}
1 & 0 & \ldots & 0 & 0 & rv_{11} & rv_{12} & \ldots & rv_{1,n-k} \\
0 & 1 & \ldots & 0 & 0 & rv_{21} & rv_{22} & \ldots & rv_{2,n-k} \\
\cdot & & & & & & & & \\
0 & 0 & \ldots & 1 & 0 & rv_{n1} & rv_{n2} & \ldots & rv_{n,n-k} \\
0 & 0 & \ldots & 0 & 1 & rz_1 & rz_2 & \ldots & rz_{n-k} \\
0 & 0 & \ldots & 0 & 0 & rq & 0 & \ldots & 0 \\
0 & 0 & \ldots & 0 & 0 & 0 & rq & \ldots & 0 \\
\cdot & & & & & & & & \\
0 & 0 & \ldots & 0 & 0 & 0 & 0 & \ldots & rq
\end{pmatrix}$$

The vector $\mathbf{y}^* = (y_1, \ldots, y_n, 0, 0, \ldots, 0)$ is a vector in the lattice and has at most t nonzero entries. If $r \geq t$, then $\mathbf{y}^*$ will be the shortest vector in the lattice. (Since K generates a t-error correcting linear code, there cannot be two vectors, $\mathbf{y}_1^*$ and $\mathbf{y}_2^*$, such that $K\mathbf{y}_1^* = K\mathbf{y}_2^*$ and such that Hamming weight of each $\mathbf{y}_1^*$ and $\mathbf{y}_2^*$ is $\leq t$.) Although the lattice basis reduction algorithm is not guaranteed to find y^*, this attack does cast suspicion on the security of the cryptosystem.

3.3 Goodman-McAuley Knapsack Cryptosystem

The Goodman-McAuley [61] knapsack cryptosystem uses modular multiplication to disguise an easy knapsack that is substantially different from those discussed above.

The Goodman-McAuley knapsack cryptosystem:

- Public key: Integers $b_1, \ldots, b_n$, q and p.
- Private key: Integers h, r satisfying $h \geq r + q$; primes $p_1, \ldots, p_n$ such that $p_i \geq 2^h$ for $1 \leq i \leq n$; $p = \Pi_{i=1}^n p_i$; non-negative integers a_{ij} for $1 \leq i, j \leq n$, such that $\Sigma_{j=1}^n a_{ij} < 2^r$ for $1 \leq i \leq n$, and the matrix $A = (a_{ij})$ is nonsingular; non-negative integers a_i such that $a_i \equiv a_{ij} \bmod p_j$ for $1 \leq i \leq n$, $1 \leq j \leq n$; W relatively prime to p such that $b_i \equiv Wa_i \bmod p$, for $1 \leq i \leq n$.
- Messages: $\mathbf{m} = (m_1, \ldots, m_n)$ such that $0 \leq m_i \leq 2^q$
- Encryption: $c = \Sigma_{i=1}^n m_i b_i \bmod p$.
- Decryption: Let $\mathbf{d} = (d_1, \ldots, d_n)$ where $d_i \equiv cW^{-1} \bmod p_i$. Then $\mathbf{m} = \mathbf{d}A^{-1}$.

If n is small, this cryptosystem can be broken by the Lenstra [97] or Kannan [78] linear programming algorithms. There is also a GCD attack that works for small r, and enables the cryptanalyst to recover all the secret information. Since $p_j \mid a_i - a_{ij}$, we have $p_j \mid b_i - a_{ij}W$, and so $p_j \mid a_{ij}b_k - a_{kj}b_i$. Since a_{kj}, $a_{ij} < 2^r$, the cryptanalyst can find (y, z) such that $p_j \mid \mathrm{GCD}(p,\ yb_k - zb_i)$ by checking all pairs y, z with $0 \leq y, z \leq 2^r$. If for each pair j, l there exist $0 \leq y, z \leq 2^r$ such that $p_j \mid yb_k - zb_i$ and $p_l \mid yb_k - zb_i$, then the cryptanalyst will find all of the p_j's by taking GCDs. If this is not the case, he can pick a different k and continue. (Note that if $p_jp_l \mid y_1b_k - z_1b_i$ and $p_jp_l \mid y_2b_k - z_2b_i$, then $y_1z_2 \equiv y_2z_1$, mod p_jp_l. Since $0 < y_1$, $z_2 < p_jp_l$, this implies $y_1z_2 = y_1z_1$. Hence if the GCD attack fails to separate p_j and p_l, then it must be the case that $\frac{a_{ij}}{a_{ki}} = \frac{a_{il}}{a_{kl}}$. But this cannot be true for all k since A is nonsingular.)

Even if n and r are chosen large enough so that the above attacks will not work, the cryptosystem can still be broken using lattice basis reduction. Consider the lattice spanned by the rows $\mathbf{v}_0, \ldots, \mathbf{v}_n$ of the matrix

$$\begin{pmatrix} b_1 & b_2 & \ldots & b_n & \varepsilon \\ p & 0 & \ldots & 0 & 0 \\ 0 & p & \ldots & 0 & 0 \\ \cdot & & & & \\ \cdot & & & & \\ \cdot & & & & \\ 0 & 0 & \ldots & p & 0 \end{pmatrix}$$

Let k_{ij} be the integers satisfying $b_iW^{-1} - k_{ij}p_j = a_{ij}$. Then $b_iW^{-1}\frac{p}{p_j} - k_{ij}p = a_{ij}\frac{p}{p_j}$. Thus, there are n vectors in the lattice of the form $W^{-1}\frac{p}{p_j}\mathbf{v}_0 - \Sigma_{j=1}^n k_{ij}\mathbf{v}_j =$

$(x_1, \ldots, x_n)$ where $\Sigma_{i=1}^{n} x_i < 2^{-q}p$. Even if the Lovasz basis reduction algorithm does not produce these vectors, but instead finds vectors $\mathbf{u}_i = (x_{i1}, \ldots, x_{in})$ that satisfy $\Sigma_{j=1}^{n} |x_{ij}| < 2^{-q}p$, the cryptanalyst can still use these vectors to break the system.

3.4 Pieprzyk Knapsack Cryptosystem

Pieprzyk [124] designed a knapsack-type cryptosystem based on polynomials over GF(2). In the following description, all polynomials are over GF(2).

The Pieprzyk Knapsack Cryptosystem:

- Public key: Polynomials $k_1(x), \ldots, k_n(x)$ and integer d.
- Private key: Polynomials $\Psi(x)$, $\phi_1(x), \ldots, \phi_n(x)$, $p_1(x), \ldots, p_n(x)$, $a(x)$ such that for $1 \leq i \leq n$, deg $\phi_i(x) = d + 1$, $\phi_i(x)$ is irreducible, $p_i(x) \equiv 1 \bmod \phi_i(x)$, $p_j(x) \equiv 0 \bmod \phi_i(x)$ if $i \neq j$, deg $\Psi(x) \geq \Sigma_{i=1}^{n} \deg\phi_i(x) + d$ and $\Psi(x$ is irreducible, and $k_i(x) \ \equiv p_i(x)a(x) \bmod \Psi(x)$.
- Messages: $M = (m_1(x), \ldots, m_n(x))$ where deg $m_i(x) \leq d$.
- Encryption: $c(x) = \ \Sigma_{i=1}^{n} m_i(x)k_i(x)$.
- Decryption: Let $c'(x) = c(x)a^{-1}(x) \bmod \Psi(x)$. $c'(x) \equiv \Sigma_{i=1}^{n} m_i(x)p_i(x) \bmod \Psi(x)$. Since deg $(\Sigma_{i=1}^{n} m_i(x)p_i(x))$ deg $< \Psi(x)$, then $c'(x) = \Sigma_{i=1}^{n} m_i(x)p_i(x)$. So $m_j(x) = c'(x) \bmod \phi_j(x)$.

The Pieprzyk knapsack cryptosystem is similar to the Goodman-McAuley knapsack cryptosystem except that integers have been replaced by polynomials. It can be broken by a similar GCD attack as well. However, in this case a much simpler solution is available. As is the case with the Luccio-Mazzone system described in Section 6.5, this cryptosystem can be broken by simple linear algebra. Note that the encryption does not involve any modular reductions. In fact, encryption is a linear transformation of the plaintext, and the matrix giving this transformation can be constructed easily from the coefficients of the polynomials $k_i(x)$. Since decryption is guaranteed to work, this matrix must have full *rank.*

3.5 Chor-Rivest Knapsack Cryptosystem

The Chor-Rivest knapsack cryptosystem [32] is the only knapsack cryptosystem that has been published that does not use some form of modular multiplication to disguise an easy knapsack. There is no feasible method known for breaking this system.

The Chor-Rivest cryptosystem:

- Public key: Integers $c_0, c_1, \ldots, c_{p-1}, p, h$, where p is a prime power, $h \leq p$, and finding discrete logarithms in GF(p^h) is feasible.
- Private key: $f(x)$ is a monic irreducible polynomial over GF(p) of degree h, GF(p^h) will be implemented as GF$(p)[x]/f(x)$, t is a root of $f(x)$, g is a generator of the multiplicative group of GF(p^h); for $\alpha \in$ GF(p), a_α is an integer such that $g^{a_\alpha} = t + \alpha$, π is a one-to-one map from $\{0, 1, \ldots, p - 1\}$ into GF(p),$b_i = a_{\pi(i)}$, d is an integer, $0 \leq d \leq p^h - 2$, and $c_i = b_i + d$.

- Messages: Vectors $M = (m_0, \ldots, m_{p-1})$ of non-negative integers such that $\Sigma_{i=0}^{p-1} m_i = h$.
- Encryption: $E(M) \equiv \Sigma_{i=0}^{p-1} m_i c_i \bmod p^h - 1$.
- Decryption: Compute $r \equiv E(M) - hd \equiv \Sigma_{i=0}^{p-1} m_i b_i \bmod p^h - 1$. Then $g^r = \Pi_{i=0}^{p-1} g^{m_i b_i}$. Since we are implementing GF(p^h) as GF$(p)[x]/f(x)$, g^r is represented as a polynomial in x of degree $< h$. Now $\theta(t) = \Pi_{j=0}^{p-1} (t + \pi(j))^{m_j}$ is represented by a polynomial of degree h, and $\theta(x) = g^r$ in GF$(p)[x]/f(x)$. So $\theta(x) = u(x) + f(x)$ in GF$(p)[x]$, where $u(x)$ represents g^r. Thus by factoring $u(x) + f(x)$, the values of $m_0, \ldots, m_{p-1}$ can be obtained.

Chor and Rivest show that if some of the secret information is revealed, then the

d are known in

some model of GF(p^h), or if t is known, or if π and d are known. They also mention an attack with nothing known that runs in $O(p^{2\sqrt{h}} h^2 \log p)$. However, this attack is infeasible for the parameters they suggest (e.g., $p = 197$ and $h = 24$).

d is known, then it is possible to reduce the cryptanalysis problem to the problem of finding the root of a very high-degree polynomial. Since d is known, the b_i's are also known. It can be shown that one can assume that the cryptanalyst knows $\pi^{-1}(0)$ and $\pi^{-1}(1)$. Without loss of generality, suppose $\pi(0) = 0$ and $\pi(1) = 1$. Then $g^{b_0} = t$. Let $e_0 = b_0^{-1} \bmod p^h - 1$. Then $t^{e_0} = g$, and $t^{e_0 b_1} = g^{b_1} = t + 1$. A somewhat similar

d is not known. In both cases, though, one has to find the root of a very high-degree trinomial. Rabin [133] and Ben-Or [11], for example, have shown that a root of a polynomial of degree w over GF(p) can be found in $O(w \log w \log \log$

p^h. No faster method for finding a root of a trinomial is known.

4 THE ONG–SCHNORR–SHAMIR (OSS) SIGNATURE SCHEMES

Ong, Schnorr, and Shamir [121] proposed a signature scheme based on polynomial equations modulo n. Their motivation was to develop a scheme that requires little computation for generation and verification of signatures, an area where the RSA scheme is deficient.

- Public key: Polynomial $P(x_1, \ldots, x_d)$ and modulus n.
- Private key: A method of solving $P(x_1, \ldots, x_d) \equiv m \bmod n$ for $x_1, \ldots, x_d$ using only a small number of multiplications, additions, and divisions mod II.
- Messages: $m \in \mathbb{Z}_n$.
- Signature: $x_1, \ldots, x_d$ such that $P(x_1, \ldots, x_d) \equiv m \bmod n$.
- Verification: Check that $P(x_1, \ldots, x_d) \equiv m \bmod n$.

This scheme generated a great deal of interest when it was first announced [120] using a polynomial P of degree 2. In fact the authors offered \$100 reward for its cryptanalysis. This reward was won **by** Pollard [127], but this did not deter the authors, who within a few months described a cubic version. This was also broken by Pollard [127], which caused the authors to publish a quartic version [122]. This version was broken by

Previous Chapter Previous Page Home Next page Next Chapter

Estes, Adleman, Kompella, McCurley, and Miller [53] and independently by Schnorr [127], and there have been no more schemes of this type proposed. We will give a brief exposition of the Pollard attack on the quadratic version.

4.1 Cryptanalysis ot the Quadratic OS5 Signature Scheme

The quadratic version proposed in [120] uses the polynomial $P(x_1, x_2) = x_1^2 + kx_2^2$ where the private key is an integer u such that $k = u^2$. To forge a signature to m, it is necessary to find x, y such that $x^2 + ky^2 = m$.

Note that the signature scheme is multiplicative, that is, if $x_1^2 + ky_1^2 = m_1$ and $x_2^2 + ky_2^2 = m_2$, then $x = x_1x_2 - ky_1y_2$ and $y = x_1y_2 + x_2y_1$ is a solution to $x^2 + ky^2 = m_1m_2$.

The Pollard algorithm:

1. Do steps (2) and (3), below, until m and k are small enough so that $x^2 + ky^2 = m$ can be solved with $x, y \in \{0, 1\}$, or until m is a square.
2. Replace m by a number $< 2\sqrt{k}$.
3. Interchange m and k by using $x \leftarrow \frac{x}{y}$ and $y \leftarrow \frac{1}{y}$.
4. Solve with $x, y \in \{0, 1\}$ and use the transformations of (2) and (3) to work back to the original equation.

To explain step (2), first find m_0 such that $m_0 = m \bmod n$, $m_0 \equiv 3 \bmod 4$, m_0 is prime, and $-k$ is a quadratic residue mod m_0. The integer m_0 is found by examining the integers in the sequence m, $m + n$, $m + 2n$, until an integer is found that satisfies all the conditions. Assuming appropriate randomness conditions on this set of integers, a success is expected with $O(\log\ n)$ trials. Solve $x_0^2 \equiv -k \bmod m_0$ and thus $x_0^2 + k = m_0m_1$.

Next we want to find $m_2 < 2\sqrt{k}$ and x_1, y_1 such that $x_1^2 + y_1^2\, k = m_1m_2$. Let $Q = m_1^{1/2} k^{-1/4}$. Use continued fractions to find a, b with $|a| < Q$ such that $\left|\frac{x_0}{m_1} + \frac{b}{a}\right| \leq \frac{1}{aQ}$. Setting $x_1 = x_0a + m_1b$ and $y_1 = a$ satisfies the requirements.

Using the multiplicative property, the problem of solving $x^2 + ky^2 \equiv m \bmod n$ reduces to solving $x^2 + ky^2 \equiv m_2 \bmod n$ and this satisfies step (2).

$O(\log\ \log\ k)$ iterations of steps (2) and (3) will result in a small enough k and m so that a solution is easily found.

4.2 Other OSS Schemes

After Pollard broke the quadratic and cubic OSS schemes, Ong, Schnorr, and Shamir developed a scheme using fourth-degree polynomials which was essentially a quadratic scheme over a quadratic number field. This scheme was broken [53] by reducing its cryptanalysis to the cryptanalysis of the quadratic scheme over the integers.

The success that the cryptanalysts have had with the OSS schemes does not imply that there are no secure signature schemes of this type. However, it is enough evidence to create strong suspicions about the security of any such schemes. Also, as their complexity increases, their speed improvement over the RSA and Rabin schemes decreases. For these reasons, there has been no further search for new OSS-type signature schemes.

5 THE OKAMOTO-SHIRAISHI SIGNATURE SCHEME

The Okamoto-Shiraishi signature scheme [119] is based on the difficulty of finding approximate *kth* roots mod *n*. This signature scheme is interesting because (as is case with the OSS schemes) it is possible to generate these signatures much faster than RSA signatures. It has also been used as an example of a subliminal channel [154].

- Private key: Factorization of $n = p^2q$.
- Public key: n, a small integer k, and a one-way function h.
- Messages: m in the domain of h.
- Signature: s such that $s^k - h(m) \equiv \delta \bmod n$ where $|\delta| \leq n^{2/3}$.

This scheme was originally proposed for $k = 2$. This version was quickly broken by Brickell and DeLaurentis [24]. The techniques of this attack also extend to $k = 3$. Shamir [148] found a different method to break the $k = 2$ case. We will present his method and also the Brickell and DeLaurentis [24] method for $k = 3$.

For $k = 2$, the forger is given $M = h(m)$ and wants to find s such that $s^2 - M \equiv \delta \bmod n$ for some δ satisfying $|\delta| \leq n^{2/3}$. The forger picks r and computes x such that $1 \leq x < n^{1/3}$ and $2rx - M + r^2 \equiv \gamma \bmod n$ for $\gamma < O(n^{2/3})$. Such an x does not exist for all choices of r (e.g., $r = (n + 1)/2$). However if the $n^{2/3}$ different valid signatures to M mod n are randomly distributed over the interval $[0, n]$, then we expect that an x will exist for most choices of r. If an x exists, it can be found through a variation of the extended Euclidean algorithm ([119] middle bits methods). Given x, $s = r + x$ is then a valid signature.

For the cubic scheme, again let $M = h(m)$. Pick $r = [n/3]$ (i.e., $r = (n/3) + \theta$ for $|\theta| \leq 1/2$.) Compute $z = M - r^3 \bmod n$, and let x = nearest integer to $z^{1/3}$ that is divisible by 3 (i.e., $x = z^{1/3} + \varepsilon$ for $|\varepsilon| \leq 3/2$). Then $s = r + x$ is a valid signature to m, since

$$\begin{aligned} s^3 &\equiv r^3 + 3r^2x + 3rx^2 + x^3 \bmod n \\ &\equiv \mathrm{r}^3 + 3\left(\tfrac{n}{3} + \theta\right)^2 x + 3\left(\tfrac{n}{3} + \theta\right) x^2 + z + 3z^{2/3}\,\varepsilon + 3z^{1/3}\,\varepsilon^2 + \varepsilon^3 \bmod n \\ &\equiv M + 3\theta^2x + 3\theta x^2 + 3z^{2/3}\,\varepsilon + 3z^{1/3}\,\varepsilon^2 + \varepsilon^3 \bmod n \\ &\equiv M + \delta \bmod n \quad \text{for } |\delta| \leq O(n^{2/3}) \end{aligned}$$

Although this specific attack is easily guarded against by disallowing signatures that are close to $\frac{n}{3}$, the basic attack can be generalized. For example, let $r = [un/v]$ for an arbitrary rational $\frac{u}{v}$. Pick x as above except divisible by v^2. If ε is small enough, then $r + x$ will be a valid signature, otherwise pick a different u, v and try again.

Okamoto [117] also proposed an encryption scheme based on similar ideas. Again n is an integer of the form $n = p^2q$. The public key also contains an integer $u = a + bpq$ where $0 < a < \frac{1}{2}\sqrt{pq}$. This system can be broken by using $u^2 \bmod n$ to solve for a. We have

$$u^2 \equiv a^2 + 2abpq \equiv 2au - a^2 \bmod n$$

Solve $0 < a < n^{1/3}$ and $|u^2 - 2au \bmod n| < n^{2/3}$ by the methods mentioned earlier. After Shamir discovered how to break this scheme (his attack is discussed in [118]), Okamoto [118] modified the cryptosystem. In the new system, u is chosen in a different manner. A message (m_1, m_2) for $0 < m_i < n^{1/9}$, $i = 1, 2$ is encrypted as $c \equiv (m_1u + m_2)^l \bmod n$. Vallee, Girault, and Toffin [158,159] cryptanalyzed this modified scheme for any l by using lattice basis reduction.

6 ADDITIONAL BROKEN TWO-KEY SYSTEMS

In this section, we will discuss several two-key cryptosystems that were broken soon after their publication. Several of them relied on composition of polynomials over finite fields. In addition to the specific attacks on such schemes that are mentioned below, there are now some fairly general techniques for decomposing polynomials developed by von zur Gathen, Kozen, and Landau [59] that cast suspicion on all similar schemes.

6.1 Matsumoto-Imai Cryptosystem

The Matsumoto-Imai cryptosystem [101] uses polynomials over GF(2^m). The private key consists of secret information about the public encryption polynomial.

- Private key: $E(X) = a(b + X^\alpha)^\beta$.
- Public key: $E(X) = \Sigma_{i=0}^{2^m-2} e_i X^i$.
- Messages: M in GF(2^m).
- Encryption: C = E(M).
- Decryption: Use the private key to solve for M.

Matsumoto and Imai suggested that the Hamming weight of β should be small so the public key is not too long. Delsarte, Desmedt, Odlyzko, and Piret [45] showed that the public polynomial E(X) would have a special form and this form would actually reveal the private key (or at least something that was functionally equivalent to the private key).

6.2 Cade Cryptosystem

The Cade [28] cryptosystem also **uses** polynomials over GF(2^m), for $m = 3r$. Let $M(x) = x^{q+1}$, where $q = 2^m$.

- Private key: $T(x) = a_0x + a_1x^q + a_2x^{q^2}$, $S(x) = b_0x + b_1x^q + b_2x^{q^2}$ where S and T are chosen to be invertible. $P(x) \equiv SMT(x) \bmod (x^{q^3} - x)$.
- Public key: $P(x) = p_{00}x^2 + p_{10}x^{q+1} + p_{11}x^{2q} + p_{20}x^{q^2+1} + p_{21}x^{q^2+q} + p_{22}x^{2q^2}$.
- Messages: M in GF(2^m).
- Encryption: C = $P(M)$.
- Decryption: Use the private key to solve for M.

James, Lidl, and Niederreiter [74] have shown that the private variables $a_0, \ldots, b_2$ can be found from the public key. Cade [29] has since used similar ideas to develop a much more complicated cryptosystem.

6.3 Yagisawa

Yagisawa [164] described a cryptosystem that combined exponentiation mod p with arithmetic mod $p - 1$. Brickell [22] showed that it could be broken without finding the private key.

To construct a public key in Yagisawa's cryptosystem, a designer picks a prime p and integers k_1, k_2, A, and B such that $2 \le A$, $B \le p - 2$, $0 \le k_1 \le p - 2$, $0 \le k_2 \le p - 2$, and $\mathrm{GCD}(k_1 - k_2, p - 1) = 1$. He then picks integers β_1 and β_2 such that $\beta_1 + \beta_2 k_1 \equiv 1 \mod p - 1$ and computes $C \equiv B^{\beta_1 + \beta_2 k_2} \mod p$ and $D \equiv B^{\beta_1 + \beta_2} \mod p$.

The public key will consist of $(A, B, C, D, k_1, k_2, p)$. To encrypt a message (X_1, X_2, X_3) where $0 \le X_i \le p - 2$, one computes (Y_1, Y_2, Y_3) where $Y_1 \equiv X_1 + X_2 + X_3 \mod p - 1$, $Y_2 \equiv k_1 X_1 + k_2 X_2 + X_3 \mod p - 1$, $F \equiv B^{X_1} C^{X_2} D^{X_3} \mod p$, and $Y_3 \equiv A^F X_3 \mod p$.

The designer, given (Y_1, Y_2, Y_3), can compute $F \equiv \beta_1 Y_1 + \beta_2 Y_2 \mod p$ and hence can compute X_3, and then X_1 and X_2.

Even though the cryptanalyst does not know β_1 and β_2, he can decrypt in much the same manner because he can actually find B^{β_1} and B^{β_2}. To do this he first computes $r \equiv (k_1 - k_2)^{-1} \mod p - 1$. Since p is prime, for all X, $X^{r(k_1 - k_2)} \equiv X \mod p$. $(C^{k_1} B^{-k_2})^r \equiv \mathrm{B}^{\beta_1 (k_1 - k_2) r} \equiv B^{\beta_1} \mod p$ and $(C^{-1} B)^r \equiv B^{\beta_2 (k_1 - k_2) r} \equiv B^{\beta_2} \mod p$. Hence the cryptanalyst can also compute F.

6.4 Tsujii-Matsumoto-Kurosama-Itoh-Fujioka Cryptosystem

Tsujii, Matsumoto, Kurosama, Itoh, and Fujioka [157] have devised a public key cryptosystem in which encryption is the evaluation of some rational functions. They remark that if a certain polynomial in a small (e.g., 4) number of variables could be factored, then their system is insecure. Unfortunately, multivariate polynomials can be factored in polynomial time, as was shown by Lenstra [94] and others.

6.5 Luccio-Manone Cryptosystem

In our early discussions of knapsack cryptosystems we noted that in general, their linearity was a reason to be suspicious of them. A surprisingly large number of cryptosysterns that have been proposed either formally or informally have succumbed to attacks based on this weakness, for example, the Pieprzyk cryptosystem (see Section 3.4). As another example, Luccio and Mazzone [99] have proposed a system (which is not really a two-key system, though) for sending information simultaneously to several receivers. Each receiver i, $1 \le i \le n$, has a secret key (k_i, c_i) known only to himself and the sender, and a large prime p is public. To send message m_i to receiver i, for $1 \le i \le n$, the sender finds an $(n - 1)$-degree polynomial $f(z)$ in $\mathrm{GF}(p)$ [2] such that $f(k_i) = c_i m_i \mod p$ and broadcasts the coefficients of $f(z)$. Receiver i then obtains

$$m_i \equiv c_i^{-1}(f(k_1)) \mod p$$

As was noted by Hellman [68], this system is very insecure, as the coefficients of the polynomial $f(z)$ are a linear transformation of the messages $(m_1, \ldots, m_n)$, and so a knowledge of n or slightly more ciphertext-plaintext pairs suffices to break the system.

7 THE RSA CRYPTOSYSTEM

The cryptosystem found by Rivest, Shamir, and Adleman [138] is the best known two-key cryptosystem. A message is encrypted as $f(m) = m^e \bmod n$ where n is a composite integer that is usually chosen as the product of only two primes, p and q, and e is relatively prime to $(p - 1)(q - 1)$. Both n and e are public, while p and q have to be kept secret. If the cryptanalyst can factor n, he can decrypt messages just as easily as the intended user. With the exception of some special situations discussed below, it is not known how to break the RSA system without factoring n. However, this has not been proved, although there are some interesting results of Alexi, Chor, Goldreich, and Schnorr [5] that say that recovering even a single bit of information from an RSA ciphertext is as hard as deciphering the full message.

Since there are several very good surveys of integer factoring algorithms (e.g., Lenstra and Lenstra [92] and Pomerance [128]), we will not go into details, but will only sketch briefly how effective those algorithms are and what precautions need to be taken in choosing the parameters of an RSA cryptosystem. We will also briefly mention some recent developments that could have dramatic impact on this area.

It has long been recognized that the primes p and q which give the public modulus $n = pq$ have to be carefully chosen, so that, for example, $p - 1$, $p + 1$, $q - 1$, and $q + 1$ all have relatively large prime factors. However, it is easy to find primes that satisfy these conditions, as was shown by Williams and Schmid [163] and Gordon [63]. It has also been shown that using very small public encryption exponents is insecure. It has recently been shown that precaution must also be taken in choosing the secret exponent, d. Wiener [160] has proven that if $e < n$, and $d < n^{1/4}$, then d can be easily determined, and thus n can be factored.

Integer factorization has advanced significantly in the last decade. When RSA was invented, the largest "hard" integer (i.e., an integer that did not have many prime factors that were either small or of special form that allows them to be split off easily) that had been factored up to then was under 40 (decimal) digits in length. Right now, hard integers of over 110 digits are being factored. This progress is due to advances in both amounts of computing power that are available and theory. As far as hardware is concerned, the most striking development has been the successful implementation of factoring algorithms on networks of workstations. This work was pioneered by Caron and Silverman [30], and extended by A. Lenstra and M. Manasse [96]. In their recent factorization of a Ill-digit integer, Lenstra and Manasse used roughly the computing power of a 300 mips (million instructions per second) machine running for a year. What was remarkable about this was that this computation was accomplished in several weeks, employed machines from around the whole world, and used only spare time on them. This is in contrast to the situation a few years ago, when it seemed that one needed to have access either to supercomputers or to special-purpose machines like that proposed in [129] to factor large integers. Since every factor of 10 increase in computing power allows one to factor integers slightly over 10 decimal digits longer, and the Lenstra-Manasse implementation is relatively portable and extendible to networks with many more machines, one can expect that in the very near future, networks of workstations around some universities or industrial laboratories could be used in their idle time to factor 130-digit integers in a few weeks or months of elapsed time. In particular, it seems very likely that the RSA challenge cipher will be broken in the next year or so, since it involves factoring an integer of 129 digits. Since workstations are becoming more powerful very rapidly (much more rapidly than supercomputers, say) and computer networks are proliferating very fast, and are going to be much more easily accessible than special-purpose machines like that of Pomerance, Smith, and Tuler [129], one should not regard even 140-digit moduli as safe from present day algorithms.

While one can make fairly good projections about the development of technology and how that will affect the security of the RSA cryptosystem, it is much harder to be certain about theoretical developments. Most of the advances in factoring in the last decade have been due to new ideas, not faster machines. Then, for a while, theoretical advances slowed down. Most of the fast factoring algorithms that have been considered until recently have been shown (under various assumptions) to run in time

$$\exp\left((1 + o(1))\ ((\log n)\ (\log\log n))^{1/2}\right)$$

as $n \to \infty$ for the "hard" integers n that are of interest in cryptography. This was explained on technical grounds as being due to all these algorithms relying in one way or another on the density of so-called "smooth" integers (integers with only small prime factors). Recently, however, a new method was suggested by J. Pollard, developed further by H. Lenstra, and implemented by A. Lenstra and M. Manasse [95]. It is referred to as the number field sieve. It is very practical when it is applied to factoring so-called Cunningham integers, that is, integers n of the form

$$n = a^k \pm 1$$

where a is small and k is large. If we let

$$M(n, r) = \exp\left((r + o(1))(\log n)^{1/3}(\log\log n)^{2/3}\right)$$

then the number field sieve factors Cunningham integers n in time

$$M(n, 1.526\ .\ .\ .)$$

This algorithm is fast not only asymptotically, but also in practice, although it is quite complicated to implement, and A. Lenstra and Manasse have used it to factor Cunningham integers of about 150 decimal digits.

The number field sieve can also be extended to factor general integers. The best currently known method of doing this yields a running time estimate of

$$M(n, 2.080\ .\ .\ .)$$

Although asymptotically this is still far better than other algorithms, the point at which this method would be faster than algorithms such as the quadratic sieve appears to be in the vicinity of 150 decimal digits. On the other hand, the number field sieve is a very recent invention, and so it is likely that substantial improvements might occur which would make it practical.

One of the fascinating questions about RSA is whether it is as secure as factoring. There are several modifications and restrictions of RSA for which this has been proven (Rabin [132], Williams [162]), but it has never been shown for RSA itself. There are however, no known attacks on RSA that are faster than factoring the modulus.

Some of the protocols for using RSA have been broken. They are described in "Protocol Failures in Cryptosystems," by J. H. Moore [107] in this book.

7.1 Variations on the RSA Cryptosystem

While the basic RSA cryptosystem has resisted all attacks, that is not true for all variants of it. Kravitz and Reed [84] have proposed using irreducible binary polynomials in place of the primes p and q. $p(z)$ and are two secret irreducible polynomials over GF(2) of degrees and s, respectively, the public modulus is the polynomial $n(z) = p(z)\ q(z)$, and the public encryption exponent e is chosen to be relatively prime to $(2^r - 1)(2^s - 1)$. This system can be broken by factoring $n(z)$, which is usually

noted in [84], and more extensively by Delsarte and Piret [41] and by Gait [57], namely, that the decryption exponent is the multiplicative inverse of e modulo 1 of $(2^u - 1)(2^{t-u} - 1)$, $1 < u < t/2$, where $t = r + s$ is the degree of $n(z)$. Thus the

the public key.

8 DISCRETE EXPONENTIATION

In the seminal paper of Diffie and Hellman [48] which started two-key cryptography,

rithm. Let p a a primitive element mod Alice chooses a random integer a and Bob a random integer b. Alice sends α^a mod p to Bob. Bob sends α^b mod p to Alice. Then both can compute α^{ab} mod p. There have been numerous extensions of this basic scheme. The scheme clearly extends to finite fields [126]. Shmuely n when is composite. Miller [106] and Koblitz [80] have extended this idea of elliptic curves. El Gamal [52] devel-

The security of the discrete exponentiation cryptosystems is based on the difficulty of the discrete logarithm problem, that is, given α, β, find x such that $\alpha^x = \beta$. There have been significant advances in algorithms for finding discrete logarithms in GF(2^n), where a striking advance was made by Coppersmith [34]. These results are surveyed in [116] and [103]. With current algorithms, the complexity of finding discrete logarithms in a prime field GF(p) for a general prime p is essentially the same as the complexity of factoring an integer n of about the same size where n is the product of two approximately equal primes ([36,90]). In particular, the number field sieve can also be extended to compute discrete logs in prime fields, but so far it is only practical when the prime is a factor of a Cunningham integer [62]. However finding discrete logarithms in GF(2^k) is considerably easier.

When utilizing finite fields GF(q), whether q is prime or $q = 2^k$, it is necessary to ensure that q - 1 has a large prime factor, as otherwise it is easy to find discrete logarithms in GF(q). This restriction is similar to the need to choose the secret primes in the RSA system carefully [63, 163].

To date, there are no subexponential algorithms for finding discrete logarithms in elliptic curves.

9 THE McELIECE CRYPTOSYSTEM

In 1978, McEliece [104] introduced a two-key cryptosystem based on error correcting codes. An implementation of this scheme would be two to three orders of magnitude faster than RSA. It has two major drawbacks. The key is quite large and it increases the bandwidth. For the parameters suggested by McEliece [104], the key would have 2^{19} bits, and a ciphertext would be twice as long as a message.

Let d_H denote the Hamming distance. The following is a description of the McEliece cryptosystem for parameters n, k, t.

- Private key: G'—a $k \times n$ generator matrix for a Goppa code that can correct t errors; P—an $n \times n$ permutation matrix; S—a $k \times k$ nonsingular matrix.
- Public key: $G = SG'P$, a $k \times n$ matrix.
- Messages: k-dimensional vectors over GF(2).
- Encryption: $c = mG + z$ for z a randomly chosen n-dimensional vector over GF(2) with Hamming weight at most t.
- Decryption: Let $\mathbf{c}' = \mathbf{c}P^{-1}$. Using a decoding algorithm for the Goppa code, find m' such that $d_H(m'G, \mathbf{c}') \leq t$. Then $m = m'S^{-1}$.

McEliece suggested that for $n = 1024$, t should be 50. For $n = 2^r$, the maximum $k = 2^r - rt$.

The security of this scheme is based on the NP-completeness of the general decoding problem for linear codes [12]. The only attacks on the system so far have come from improvements in algorithms that would decode any error correcting code.

An obvious attack on this system is to pick k columns of the matrix G. Let G_k, $\mathbf{c}_k$, z_k be restrictions onto these k columns. If $z_k = 0$, then $mG_k = \mathbf{c}_k$, and m can be found by linear algebra. A given choice of k columns can be checked in k^3 operations (assuming that fast matrix multiplication is not used) to see if it gives an appropriate m. For n = 1024 and t = 50, the expected number of operations before a success is about $2^{80.7}$. However, Adams and Meijer [l] showed that for n = 1024, t = 37 is the optimum **value** based on this attack, and for this value of t, the expected number of operations is about $2^{84.1}$.

Lee and Brickell [91] modified this attack. They found that after picking k columns, it was more efficient to check if z_k had at most two 1's. Against this attack, for n = 1024 t = 38 is optimal. For t = 37 or 38, the expected number of operations is about $2^{73.4}$.

To the best of our knowledge, there have been no successful attempts to cryptanalyze this system which examined possible leakage of the structure of Goppa codes into the public key.

Rao and Nam [135] have proposed using a variant of the McEliece scheme as a single-key cryptosystem. The key consists of a matrix G' generated in exactly the same manner as in the McEliece scheme, and a set F of possible error vectors. A message m is encrypted by picking a $z \in F$ at random and forming $c = mG' + z$. Rao and Nam give two methods of selecting the set F. Hin [72] showed how to break the Rao-Nam system for one of these methods and Struik and Tilburg [156] for the other. Both of these attacks used a chosen plaintext attack in which the cryptanalyst needs $|F|$ different encryptions of a fixed message m.

The Rao-Nam system could be modified slightly by using a pseudo-random function f, and letting $z = f(m)$ so that there is only one encryption for each message m. It is not known if the above attacks could be modified so that they would also break this system.

10 CONGRUENTIAL GENERATOR3

A *congruential generator* is a method of generating a sequence $s_0, s_1, \ldots$ where s_i is computed by the recurrence

$$s_i \equiv \sum_{j=1}^{k} \alpha_j \phi_j(s_0, \ldots, s_{i-1}) \bmod m$$

Research in the last few years has uncovered serious weaknesses in using congruential

cryptanalyzing congruential generators in which the cryptanalyst knows the functions ϕ_i but not the coefficients α_i and the modulus m. We will examine these results in

The simplest congruential generator, the *linear congruential generator*, has the form

$$s_i \equiv as_{i-1} + b \bmod m$$

A *truncated congruential generator* generates a sequence $x_0, x_1, \ldots$ where x_i is the leading t bits of s_i for some sequence s_i produced by a congruential generator. Alternately, we could determine the x_i by some window of t of the bits of the s_i. Truncated

parameters a, b, and m are secret. We will examine these results in Sections 10.2 and 10.3.

ators.

10.1 Congruential Generators (Nontruncated)

We will evaluate the security of congruential generators relative to a variation of a known plaintext attack. We will assume that the cryptanalyst knows the functions ϕ_j, but does not know the coefficients α_i or the modulus m. The cryptanalyst is given $s_1, \ldots, s_{i-1}$. He tries to guess s_i. After he guesses, he is told the correct value. We will say that such an attack breaks the cryptosystem if there is a bound that is polynomial in log m and k on the running time of the attack and on the number of errors that are made by the cryptanalyst.

The cryptanalysis of congruential generators was started by Boyar [125] when she found how to break linear congruential generators. (Knuth [79] had an earlier result, but his algorithm was exponential in log m.) Boyar also showed how to break quadratic and cubic congruential generators. Lagarias and Reeds [89] then extended Boyar's result by showing that the same algorithm would break any congruential generator, where k = 1 and ϕ is a polynomial depending only on s_{i-1}. Recently, Krawczyk [85] has proven how to break any congruential generator, in which the functions ϕ_i are computable over the integers in time polynomial in log m.

Krawczyk's algorithm is only a slight modification of Boyar's and we will present it here because of its simplicity. The basic idea that Krawczyk introduces is that he does not try to find the α_j's.

Let

$$B_i = \begin{pmatrix} \phi_1(s_0, \ldots, s_{i-1}) \\ \phi_2(s_0, \ldots, s_{i-1}) \\ \cdot \\ \cdot \\ \cdot \\ \phi_k(s_0, \ldots, s_{i-1}) \end{pmatrix}$$

The first idea used by both Boyar and Krawczyk is that for all but possibly k values of i, there exist integers γ_j, $j = 1, \ldots, i$ such that $\gamma_i \neq 0$ and $\gamma_i B_i = \Sigma_{j=0}^{i-1} \gamma_j B_j$. Then $\gamma_i s_i \equiv \Sigma_{j=0}^{i-1} \gamma_j s_j \bmod m$. Thus, either s_i can be predicted (in the case that $\gamma_i s_i \equiv \Sigma_{j=0}^{i-1} \gamma_j s_j$) or a multiple of m can be computed after the correct value of s_i is given. The size of such a multiple of m will be polynomial in log m and k.

m, i. $\hat{m}$ be the current multiple of m that is known.

1. Given s_{i-1}, try to express B_i as $B_i \equiv \Sigma_{j=0}^{i-1} \gamma_j B_j \bmod \hat{m}$.
2. If (1) is successful, compute p as $p \equiv \Sigma_{j=0}^{i-1} \gamma_j s_j \bmod \hat{m}$ and if $p \neq s_i$, then replace $\hat{m}$ by GCD($\hat{m}$, $p - s_i$).

Krawczyk has shown that if $p \neq s_i$, then $\hat{m} \neq$ GCD($\hat{m}$, $p - s_i$). He also showed that for a fixed $\hat{m}$, step (1) fails at most k log $\hat{m} + 1$ times. From these results, it follows that this algorithm breaks these congruential generators in polynomial time.

10.2 linear Truncated Congruential Generators with Known Parameters

In this section, we will consider the security of truncated linear congruential generators in which the cryptanalyst knows the parameters a, b, and m. These generators were shown to be insecure by Frieze, Hastad, Kannan, Lagarias, and Shamir [55,56,67]. All of the known attacks are attacks on linear congruential generators in which some constant fraction of the bits of each s_i are used as the pseudorandom sequence. The attacks are all based on lattice basis reduction. Each of the attacks that we will describe has been proven to break certain truncated linear congruential generators. However, it has not been determined whether these attacks would also be effective against most truncated linear congruential generators.

Let s_i be a sequence generated by

$$s_i \equiv as_{i-1} + b \bmod m \qquad (10.1)$$

Let $n = \log_2 m$. For $0 < \beta < 1$ such that βn is an integer, we can write

$$s_i = x_i 2^{\beta n} + y_i \qquad (10.2)$$

so that y_i is the lower βn bits of s_i and x_i is the high-order $(1 - \beta)n$ bits of s_i.

To evaluate the security of these sequences, we will assume that the cryptanalyst knows $x_1, \ldots, x_{i-1}$, and he wants to predict x_i. For the remainder of this section, we will assume that $b = 0$, for if $b \neq 0$, we could examine the sequence $\hat{x}_i = x_i - x_{i-1}$. This sequence is essentially the truncation of the sequence $\hat{s}_i = s_i - s_{i-1}$ which is generated by $\hat{s}_i \equiv a\hat{s}_{i-1}$ mod m. If we could predict the sequence $\hat{x}_i$, then we could also predict the sequence x_i.

Let L be the lattice spanned by the vector $(m, 0, \ldots, 0)$ and by the $k - 1$ vectors

$$(a^{i-1}, 0, \ldots 0, -1, 0, \ldots 0), \text{ for } i = 2, \ldots k$$

where the -1 is in the i'th coordinate. All vectors $\mathbf{w} = (w_1, \ldots, w_k)$ in L satisfy

$$\sum_{i=1}^{k} w_i s_i \equiv 0 \bmod m \tag{10.3}$$

The attack consists of two steps. First, find a reduced basis for L, of vectors $\mathbf{w}^j$, $j = 1, \ldots, k$. We have

$$\sum_{i=1}^{k} w_i^j s_i = \sum_{i=1}^{k} w_i^j x_i 2^{\beta n} + \sum_{i=1}^{k} w_i^j y_i \tag{10.4}$$

If

$$\left| \sum_{i=1}^{k} w_i^j y_i \right| < \frac{m}{2} \tag{10.5}$$

for $j = 1, \ldots, k$, then since we know that each equation in Eq. (10.4) is 0 mod m, and we know the x_i, we get k independent equations over the integers for the s_i, $i = 1, \ldots, k$.

If the vectors in the reduced basis satisfy Eq. (10.5), then this attack will be successful.

Theorem 10.1 [55]: Let m be squarefree, $\varepsilon > 0$, and k be a given integer. There exist constants c_k and $C(\varepsilon, k)$ such that if $m > C(\varepsilon, k)$ and if $(1 - \beta)n > n(\frac{1}{k} + \varepsilon) + c_k$, then the reduced basis found by the Lovasz algorithm will satisfy Eq. (10.5) for at least $1 - O(m^{-\frac{\varepsilon}{2}})$ of the possible coefficients a.

The constant $c_k = O(k^2)$ and $C(\varepsilon, k) = e^{2k^{\alpha\alpha\varepsilon}}$ for some constant d_0. Frieze, et al. also have a similar result for m which are almost squarefree and they have proved Theorem 10.1 for $k = 3$ and any m. It is an interesting question to determine if this attack will work for $k > 3$ and m an integer that is not almost squarefree, for example, m = 2”. To prove that the attack will work in this case appears to need different proof techniques than those used in [55]. The attack that has been described will also be effective against truncated linear congruential generators in which some block of bits other than the most significant bits are used for the pseudorandom sequence. However, in this case, the algorithm is not quite as efficient, and asymptotically twice as many bits are needed to break the system.

It would also be interesting to determine whether this attack would be successful when the modulus m is not so large compared with k. This could probably be established by experimental evidence, but to our knowledge, there has been no computational experience with this algorithm.

10.3 Truncated linear Congruential Generators with Unknown Parameters

In this section, we assume that the cryptanalyst does not know the parameters a, *b*, and *m*. Boyar [16] showed that if only a few bits (O(log log m)) were truncated, then her attack would still work. Stern [155] has recently discovered an extension of the FHKLS method that will break the truncated linear congruential generators (LCGs) when a constant fraction of the bits have been truncated.

Let us first consider his algorithm when m is known. Let $\mathbf{v}_i$ be the vector $(x_{i+1} - x_i, x_{i+2} - x_{i+1}, x_{i+3} - x_{i+2})$. In part 1 of the algorithm, use the algorithm of Hastad, Just, Lagarias, and Schnorr [66] to find a short integer relation

$$\sum_{i=1}^{k} \lambda_i \mathbf{v}_i = 0$$

Let $\mathbf{w}_i$ be the vector $(s_{i+1} - s_i, s_{i+2} - s_{i+1}, s_{i+3} - s_{i+2})$. Then, let

$$\mathbf{u} = \sum_{i=1}^{k} \lambda_i \mathbf{w}_i$$

Stern has shown that if k is at least $\sqrt{6(1 - \beta) \log m}$, then for most a, $\mathbf{u}$ will be the zero vector. If $\mathbf{u}$ is the zero vector, then part 1 is successful.

If part 1 is successful, then the cryptosystem is insecure under a type of known plaintext attack. Assume that the cryptanalyst knows the values of $x_0, \ldots, x_{i-1}$, and that he is also given the h least significant bits of x_i. From this information, he is asked to predict the next bit of x_i. Stern has shown that if part 1 was successful, then out of the $(1 - \beta)n$ bits of x_i, the expected number of mistakes is only $\sqrt{6(1 - \beta) \log \frac{m}{2}}$.

Now we will consider the case when the cryptanalyst does not know a or m. Stern has shown that if part 1 is successful, then the polynomial $P(z) = \Sigma_{i=0}^{k-1} \lambda_i z^i$ satisfies $P(a) \equiv 0 \bmod m$. Stern suggests that by repeatedly using part 1, we could obtain many such polynomials and use them to determine m and a. Stern could prove that this method would work based on an assumption that involves the randomness of the polynomials *P*. Lacking a proof of this assumption, it would be interesting to also test this algorithm.

11 DES

The most remarkable news about the cryptanalysis of DES [112] is that there are no substantial attacks to mention. See Konheim's book [83] for a complete description of the algorithm. Although DES has been the U.S. standard for almost 10 years, and been the focus of many attempts at cryptanalysis, [17,49] it remains unbroken. The fastest attacks known at this time require $|K|/2$ encryptions where $|K| = 2^{56}$ is the total number of possible keys.

11.1 Cryptanalytic Attacks on Weakened DES

There has been some **success** in breaking weakened DES-like cryptosystems. Grossman and **Tuckerman [64]** showed DES could be made weak by modifying the method in which the S-boxes were used.

Another way to weaken DES is to shorten the number of rounds from the 16 that were proposed. Andelman and Reeds [7] developed a general technique for cryptanalyzing substitution-permutation cryptosystems which worked extremely well on networks with only three or four rounds. Chaum and Evertse [31] found a known plaintext attack on a six-round DES that is faster than exhaustive key search. Davies [37] exploited some nonrandom structures that he found in the S-boxes of DES that enabled him to break an eight-round DES using 2^{40} known plaintext messages. Biham and Shamir [14] have recently announced a chosen plaintext attack that can break an eight-round DES with 2^{18} chosen plaintext-ciphertext pairs. Their method extends to a 15-round DES, which can be broken with 2^{52} chosen plaintext-ciphertext pairs. However, for the full 16-round DES, their method requires more plaintext-ciphertext pairs than the 2^{55} encryptions needed for an exhaustive key search.

Although there has been no success against the full DES algorithm, there has been cryptanalytic success in breaking one of the proposed modes of operation of DES [113]. In output feedback mode (OFB), DES is used to generate a pseudorandom sequence, which is then used as a one-time pad to encrypt the message. It makes use of a function, $f_{k,m}$, where k is any valid DES key and $1 \leq m \leq 64$. $f_{k,m}(x) = x$ shifted left m bits and concatenated with the leftmost m bits of $E_k(x)$. ($E_k(x)$ is the DES encryption of x using key k). To generate a sequence $s_1, \ldots$ using OFB, a key, k, and an initial 64-bit vector x_0 are chosen. Then for $i \geq 1$, $s_i = E_k(x_{i-1})$ and $x_i = f_{k,m}(x_{i-1})$. Davies and Parkin [38] observed that for a fixed key k and $m = 64$, the function, f, is a permutation. The expected cycle size of a random permutation on N elements is $N/2$. However, if $m \leq 63$, then f is not a permutation. The expected cycle size for a random function on N elements is only about $N^{1/2}$. Therefore only $m = 64$ should be considered secure for OFB.

11.2 Cycles in DES

Kaliski, Rivest, and Sherman [77] examined DES to see if any of several properties held. As an example, they wanted to determine whether the 2^{56} permutations E_k for $k \in K$ formed a subgroup. That is, for any two keys k_1 and k_2, is there another key k_3 such that $E_{k_1}(E_{k_2}(x)) = E_{k_3}(x)$ for all messages x. It was quite important to determine if DES had these properties, because if any one of them held, there would be an attack on DES that would require only $\sqrt{|K|}$ operations. By examining the results of some cleverly designed experiments on DES, they concluded that it was extremely unlikely that DES had any of these properties.

Additional cycling experiments have been performed by Moore and Simmons [108,109]. Soon after DES was released, four keys were labeled as weak keys. (These keys had the first 28 bits identical and also the last 28 bits identical.) In addition, several other keys were labeled as semiweak keys [75]. Coppersmith [35] and Moore and Simmons found some remarkable properties of these keys. In particular, they were able to find fixed points or antifixed points, that is, messages such that the encryption of the message is either the message itself or the complement of the message. Unfortunately, it is not apparent how to apply these results to give any information about other keys.

Quisquater and Delescaille [130] constructed an algorithm for finding collisions in DES. A collision is a message, m, and a pair of keys, k_1, k_2, such that both keys encrypt the message to the same ciphertext. Using their algorithm, they discovered

Previous Chapter Previous Page Home Next page Next Chapter

many collisions in DES. It is not known how the existence of collisions can be used to aid in the cryptanalysis of DES.

11.3 Structural Properties of DES

The S-boxes introduce nonlinearity into the DES. There are eight S-boxes in the DES, each of which is a set of four permutations on sixteen elements. In the first public analysis of DES by Hellman, Merkle, Schroeppel, Washington, Diffie, and Schweitzer [69], there were several properties noted that were satisfied by all of the S-boxes. It was obvious that the S-boxes were not chosen at random, but there is no known cryptographic weakness resulting from these properties. Shamir [147] discovered an additional property of the S-boxes that at first looked very suspicious. However, Brickell, Moore, and Purtill [26] showed that this additional property was the result of the design properties noted by Hellman et al. [69] and Brickell et al. [26].

11.4 Birthday Attacks

There have been some cryptanalytic attacks based on the so called “birthday paradox.” If $\alpha\sqrt{n}$ are drawn with replacement from a set of size n, the probability that two of them will be a match is about 1 - $e^{-\alpha^2/2}$. This means that in a random group of 24 people, the probability that two will have the same birthday is about $1/2$. This is an old and well-understood concept and it has been the essential point of some recent cryptanalytic attacks.

Rabin [131] described a scheme for authenticating data using any block cipher as a hash function and RSA for a signature of the hashed value. Yuval [165] showed that this system could be broken with a birthday attack. Let n be the size of the image space of the hash function. (For DES, $n = 2^{64}$.) Suppose that Alice has two messages, x, a message that Bob wants to sign, and y, a message that Alice wants signed but Bob is not willing to sign. Alice prepares about $\sqrt{n}$ different slight variations of x and of y and computes the hash functions of each of them. With high probability, she will find variations $\hat{x}$ and $\hat{y}$ that hash to the same point. She gives $\hat{x}$ to Bob to be signed, but she can now use the signature of $\hat{x}$ as a signature for 9.

Another version of the birthday attack can be used to break this system even if Alice only has access to one valid signature and cannot obtain any additional ones. In the Rabin scheme, a text, $M_1, \ldots, M_r$, is signed by picking an H_0 at random. Then, $H_i = E_{M_i}(H_{i-1})$ for $i = 1, \ldots, r$. Finally, the pair (H_0, H_r) is signed using RSA.

Suppose that Alice is given the signature for a pair (H_0, G). She then picks $M_1, \ldots, M_{r-2}$ to be anything she likes, and computes $H_i = E_{M_i}(H_{i-1})$ for $i = 1, \ldots, r - 2$. Alice picks $\sqrt{n}$ X's and computes $E_X(H_{r-2})$ for each X. She picks $\sqrt{n}$ Y's and computes $D_Y(G)$ for each Y. With high probability, she will find a pair (X, Y) such that $E_X(H_{r-2} = D_Y(G)$. Then the signature for (H_0, G) will be a valid signature for $M_1, \ldots, M_{r-2}, X, Y$.

Davies and Price [39] proposed an iterated form of the Rabin scheme to avoid this latter birthday attack. They proposed going through all of the messages twice. Coppersmith [33] showed that this scheme is still susceptible to a birthday attack. See Jueneman [76] for a survey of these results.

12 FAST DATA ENCIPHERMENT ALGORITHM

The fast data encipherment algorithm (FEAL) was proposed by Shimizu and Miyaguchi [152] at Eurocrypt '87 as an alternative to DES for use in software. FEAL is a four-round substitution-permutation cryptosystem with a 64-bit key. Den Boer [42] soon found an attack on FEAL that requires only 10,000 chosen plaintexts. This has since been improved by Murphy [1 10] to an attack which needs only 20 chosen plaintexts. FEAL has since been modified to become FEAL-N [111], where N is the number of rounds. The methods that Biham and Shamir [14] developed can be used to break FEAL-8 with less than 2000 chosen plaintexts, and to break FEAL-N for N ≤ 31 with fewer chosen plaintexts than the number of encryptions needed in an exhaustive key search.

13 ADDITIONAL COMMENTS

In this section we will mention a few additional results, but without any details. The need to protect computer files has created a need for very efficient secure cryptosysterns. However, many of the cryptosystems designed and sold to fill this need have been shown to be insecure. Reeds and Weinberger [136] have shown how to break the UNIX crypt command using a ciphertext-only attack. Kochanski [82] studied five security products designed for the IBM personal computer. He found them to be extremely insecure. He broke all of them using only a PC and without any knowledge about the encryption algorithms that was not provided by the manufacturer with the purchase of the product. He broke four of them with a ciphertext-only attack. A purchaser of these products should be very skeptical about their claims of security.

Rivest, Adleman, and Dertouzos [137] introduced several privacy homomorphisms. Essentially, a *privacy homomorphism* is an encryption function in which desired operations on plaintext messages can be achieved by performing corresponding operations on ciphertext messages. For example, $E(m) \equiv m^e \bmod n$, the RSA encryption function, is a privacy homomorphism since $E(m_1) * E(m_2) \equiv E(m_1m_2)$ mod II. There are four other privacy homomorphisms mentioned in [137]. Brickell and Yacobi [27] showed that two of these can be broken with ciphertext-only attacks and the other two can be broken with known plaintext attacks.

Although many of the encryption machines used during World War II were broken during the war, new techniques for breaking them are still being discovered. The techniques of Andelman and Reeds [7] for cryptanalyzing rotor machines and the comprehensive book covering cryptanalysis of World War II-era encryption machines by Deavours and Kruh [40] are excellent examples.

Schnorr [142] proposed an algorithm for constructing a string, G,(x) of length $2n2^{2n}$ bits from a random seed, x, of length $n2^n$ bits. He claimed that no statistical test that depended on fewer than $2^{o(n)}$ bits could distinguish G,(x) from a random bit string. However, Rueppel [139] has demonstrated a statistical test that depends on only $4n$ bits that does distinguish (with very high probability) G,(x) from a random string. Furthermore, Rueppel has shown that the seed, x, can be computed in time $O(n2^n)$ using only $n2^n$ + O(1) bits of G,(x). Thus, Schnorr's random number generator expands the randomness of the seed by at most a constant number of bits.

Matyas and Shamir [100] developed a novel idea for encrypting video signals. A randomly generated curve that passes through all pixels of a video signal is used to transmit the video picture. The light values at the pixels are then sent in the clear. Hastad [65] showed that this method was insecure if the same curve is used to transmit many pictures. Bertilsson, Brickell, and Ingemarsson [13] showed that it was insecure if many different curves were used to transmit similar pictures. Together, these results indicate that this scheme is unlikely to be secure without some major modifications.

A radically different concept for a cryptosystem has been proposed by Bennett, Brassard, Breidbart, and Wiesner [10]. They call it quantum cryptography and its security is based on the uncertainty principle of quantum physics. (A very complete list of references on this subject can be found in the report of Bennett and Brassard [9].) If such systems become feasible, the cryptanalytic tools discussed here will be of no use.

ACKNOWLEDGMENTS

The authors would like to thank Joan Boyar, D. Coppersmith, J. Hastad, H. W. Lenstra, Jr., and C. Schnorr for useful comments.

REFERENCES

[1] C. M. Adams and H. Meijer, "Security-related comments regarding McEliece's public-key cryptosystem," in *Lecture Notes in Computer Science 293; Advances in Cryptology: Proceedings of Crypto* '87, C. Pomerance, Ed., Santa Barbara, CA, Aug. 16-20, 1987, pp. 224-230. Berlin: Springer-Verlag, 1988.

[2] B. S. Adiga and F? Shankar, "Modified Lu-Lee cryptosystem," *Electron. Lett.*, vol. 21, no. 18, pp. 794-795, Aug. 29, 1985.

[3] L. M. Adleman, "On Breaking Generalized Knapsack Public Key Cryptosystems," *Proc. 15th ACM Symposium on Theory of Computing*, pp. 402-412, 1983.

[4] L. M. Adleman and R. L. Rivest, "How to break the Lu-Lee (COMSAT) public-key cryptosystem," MIT Laboratory for Computer Science, July 1979.

[5] W. Alexi, B. Chor, 0. Goldreich, and C. I? Schnorr, "RSA and Rabin functions: Certain parts are as hard as the whole," *SIAM J. Comput.*, vol. 17, pp. 194-209, 1988.

[6] H. R. Amirazizi, E. D. Karnin, and J. M. Reyneri, "Compact knapsacks are polynomially solvable," extended abstract in *Crypto '81 Abstracts*, IEEE Workshop on Communications Security, Santa Barbara, CA, Aug. 24-26, 1981. Reprinted in *ACM SIGACT NEWS*, vol. 15, pp. 20-22, 1983.

[7] D. Andelman and J. Reeds, "On the cryptanalysis of rotor machines and substitution-permutation networks," *IEEE Trans. Inform. Theory*, vol. IT-28, no. 4, pp. 578-584, July 1982.

[8] B. Arazi, "A trapdoor multiple mapping," *IEEE Trans. Inform. Theory*, vol. 26, no. 1, pp. 100-102, Jan. 1980.

[9] C. H. Bennett and G. Brassard, "Quantum public key distribution reinvented," *SIGACT* News, vol. 18, no. 4, pp. 51-53, Summer 1987.

[10] C. H. Bennett, G. Brassard, S. Breidhart, and S. Wiesner, "Quantum cryptography, or unforgeable subway tokens," in Advances *in Cryptology: **Proc**. Crypto* '82, D. Chaum, R. L. Rivest, and A. **T.** Sherman, Eds., Santa Barbara, CA, Aug. 23-25, 1982, pp. 267-275. New York: Plenum Press, 1983.

[11] M. Ben-Or, "Probabilistic algorithms in finite fields," in ***Proc**. 22nd IEEE Found. Computer Sci. Symp.*, pp. 394-398, 1981.

[12] E. R. Berlekamp, R. J. **McEliece,** and H. C. A. van Tilborg, "On the inherent intractability of certain coding problems," *IEEE Trans. Inform. Theory,* vol. **IT-**24, pp. 384-386, 1978.

[13] M. Bertilsson, E. F. Brickell, and I. Ingemarsson, "Cryptanalaysis of video encryption based on space-filling curves," in *Lecture Notes in Computer Science 434; Advances in Cryptology; **Proc**. Eurocrypt* '89, J.-J. Quisquater and J. **Vandewalle,** Eds., Houthalen, Belgium, April 10-23, 1989, pp. 403-411. Berlin: Springer-Verlag, 1990.

[14] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosysterns," extended abstract in *Advances in Cryptology: **Proc**. Crypto '90, S.* **Vanstone**, Ed., Santa Barbara, CA, Aug. 11-15, 1990. Berlin: Springer-Verlag (in press) to appear in *1. Cryptology,* vol. 4, no. 1, 1991.

[15] J. Boyar, "Inferring sequences produced by pseudo-random number generators," *J. ACM,* vol. 36, pp. 129-141, 1989.

[16] J. Boyar, "Inferring sequences produced by a linear congruential generator missing low-order bits," ***J.** Cryptology,* vol. 1, no. 3, pp. 177-184, 1989.

[17] D. K. Branstead, J. Gait, and S. Katzke, "Report of the workshop on cryptography in support of computer security," *NBSIR* 77-1291, National Bureau of Standards, September 21-22, 1976, Sept. 1977.

[18] G. Brassard, "A note on the complexity of cryptography," *IEEE Tran. lnform. Theory,* vol. IT-25, pp. 232-233, 1979.

[19] E. **F.** Brickell, "Solving low density knapsacks," in *Advances in Cryptology: **Proc**. Crypto '83,* D. Chaum, Ed., Santa Barbara, CA, Aug. 22-24, 1983, pp. 25-37. New York: Plenum Press, 1984.

[20] E. **F.** Brickell, "A new knapsack based cryptosystem," paper presented at Crypto '83, Santa Barbara, CA, Aug. 22-24, 1983.

[21] E. F. Brickell, "Breaking iterated knapsacks," in *Lecture Notes in Computer Science 196; Advances in Cryptology: **Proc**. Crypto* '84, G. R. Blakley and D. Chaum, Eds., Santa Barbara, CA, Aug. 19-22, 1984, pp. 342-358. Berlin: Springer-Verlag, 1985.

[22] E. **F.** Brickell, "Cryptanalysis of the Yagisawa public key cryptosystem," *Abstracts **of** Papers,* Eurocrypt '86, May 20-22, 1986.

[23] E. F. Brickell, "The cryptanalysis of knapsack cryptosystems," in *Applications **of** Discrete Mathematics,* R. D. Ringeisen and **F.** S. Roberts, Eds., pp. 3-23. Philadelphia: Society for Industrial and Applied Mathematics, 1988.

[24] E. F. Brickell and J. M. **DeLaurentis,** "An attack on a signature scheme proposed by Okamoto and Shiraishi," in *Lecture Notes in Computer Science 218; Advances in Cryptology: **Proc**.* **Crypto** '85, H. C. Williams, Ed., Santa Barbara, CA, Aug. 18-22, 1985, pp. 28-32. Berlin: Springer-Verlag, 1986.

[25] E. F. Brickell, J. C. Lagarias, and A. M. Odlyzko, "Evaluation of the Adelman attack on multiple iterated knapsack cryptosystems," in *Advances in Cryptology:*

Proc. Crypto '83, D. Chaum, Ed., Santa Barbara, CA, Aug. 22-24, 1983, pp. 39-42. New York: Plenum Press, 1984.

[26] E. F. Brickell, J. H. Moore, and M. R. Purtill, "Structure in the S-boxes of the DES," extended abstract in *Lecture Notes in Computer Science 263; Advances in Cryptology: Proc. Crypto* '86, A. M. Odlyzko, Ed., Santa Barbara, CA, Aug. 11-15, 1986, pp. 3-8. Berlin: Springer-Verlag, 1987.

[27] E. F. Brickell and Y. Yacobi, "On privacy homomorphisms," in *Lecture Notes in Computer Science 304; Advances in Cryptology: Proc. Eurocrypt '87,* D. Chaum and W. L. Price, Eds., Amsterdam, The Netherlands, April 13-15, 1987, pp. 117-126. Berlin: Springer-Verlag, 1988.

[28] J. J. Cade, "A public key cipher which allows signatures," paper presented at 2nd SIAM Conference on Applied Linear Algebra, Raleigh, NC, 1985.

[29] J. J. Cade, "A modification of a broken public-key cipher," in *Lecture Notes in Computer Science 263; Advances in Cryptology: Proc. Crypto '86,* A. M. Odlyzko, Ed., Santa Barbara, CA, Aug. 11-15, 1986, pp. 64-83. Berlin: Springer-Verlag, 1987.

[30] T. R. Caron and R. D. Silverman, "Parallel implementation of the quadratic scheme," *J. Supercomputing,* vol. 1, no. 3, pp. 273-290, 1987.

[31] D. Chaum and J. Evertse, "Cryptanalysis of DES with a reduced number of rounds," in *Lecture Notes in Computer Science 218; Advances in Cryptology: Proc. Crypto* '85, H. C. Williams, Ed., Santa Barbara, CA, Aug. 18-22, 1985, pp. 192-211. Berlin: Springer-Verlag, 1986.

[32] B. Chor and R. Rivest, "A knapsack type public key cryptosystem based on arithmetic in finite fields," *IEEE Trans. Inform. Theory, vol. 34,* pp. 901-909, 1988.

[33] D. Coppersmith, "Another birthday attack," in *Lecture Notes in Computer Science 218; Advances in Cryptology: Proc. Crypto* '85, H. C. Williams, Ed., Santa Barbara, CA, Aug. 18-22, 1985, pp. 14-17. Berlin: Springer-Verlag, 1986.

[34] D. Coppersmith, "Fast evaluation of logarithms in fields of characteristic two," *IEEE Trans. Inform. Theory,* vol. IT-30, pp. 587-594, 1984.

[35] D. Coppersmith, "The real reason for Rivest's phenomenon," in *Lecture Notes in Computer Science 218; Advances in Cryptology: Proc. Crypto '85,* H. C. Williams, Ed., Santa Barbara, CA, Aug. 18-22, 1985, pp. 535-536. Berlin: Springer-Verlag, 1986.

[36] D. Coppersmith, A. M. Odlyzko, and R. Schroeppel, "Discrete Logarithms in *GF*(p),'' *Algorithmica,* vol. 1, no. 1, pp. 1-16, 1986.

[37] D. W. Davies, "Investigation of a potential weakness in the DES algorithm," unpublished manuscript circulated in July 1988.

[38] D. W. Davies and G. I. P. Parkin, "The average cycle size of the key stream in output feedback encipherment," in *Advances in Cryptology: Proc. Crypto '82,* D. Chaum, R. L. Rivest, and A. T. Sherman, Eds., Santa Barbara, CA, Aug. 23-25, 1982, pp. 97-98. New York: Plenum Press, 1983.

[39] D. W. Davies and W. L. Price, "The application of digital signatures based on public key cryptosystems," *NPL Report DNACS 39/80,* National Physical Laboratory, Teddington, Middlesex, England, Dec. 1980.

[40] C. A. Deavours and L. Kruh, *Machine Cryptography and Modern Cryptanalysis,* Dedham, MA: Artech House, 1985.

[41] I? Delsarte and P. Piret, “Comment on ‘Extension of RSA cryptostructure: A Galois approach’ ,’’ *Electron. Lett.,* vol. 18, pp. 582-583, 1982.

[42] B. Den Boer, “Cryptanalysis of F.E.A.L.,” in *Lecture Notes in Computer Science 330; Advances in Cryptology: Proc. Eurocrypt* ‘88, C. G. Gunther, Ed., Davos, Switzerland, May 25-27, 1988, pp. 293-299. Berlin: Springer-Verlag, 1988.

[43] D. E. R. Denning, *Cryptography and Data Security,* Menlo Park, CA: Addison-Wesley, 1983.

[44] Y. Desmedt, “What happened with knapsack cryptographic schemes?” in *Performance Limits in Communication, Theory and Practice,* J. K. Skwirzynski, Ed., Boston/Dordrecht/London: Kluwer, pp. 113-134, 1988.

[45] P. Delsarte, Y. Desmedt, A. Odlyzko, and I? Piret, “Fast cryptanalysis of the Matsumoto-Imai public key scheme,” in *Lecture Notes in Computer Science 209; Advances in Cryptology: Proc. Eurocrypt* ‘84, T. Beth, N. Cot, and I. Ingemarsson, Eds., Paris, France, April 9–11, 1984, pp. 142-149. Berlin: Springer-Verlag, 1985.

[46] Y. G. Desmedt, J. P. Vandewalle, and R. J. M. Govaerts, “A critical analysis of the security of knapsack public key algorithms,” *IEEE Trans. Inform. Theory,* vol. IT-30, no. 4, pp. 601-611, July 1984. Also in *Abstract of Papers, IEEE Intern. Symp. Inform. Theory* (Les Arcs, France), pp. 115-116, June 1982.

[47] Y. Desmedt, J. Vandewalle, and R. Govaerts, “A general public key cryptographic knapsack algorithm based on linear algebra,” in Abstract *of Papers, IEEE Intern. Symp. Inform. Theory,* St. Jovite, Quebec, Canada, Sept. 26-30, 1983, pp. 129-130, 1983.

[48] W. Diffie and M. E. Hellman, “New directions in cryptography,” *IEEE Trans. Inform. Theory,* vol. IT-22, pp. 644-654, 1976.

[49] W. Diffie and M. E. Hellman, “Exhaustive cryptanalysis of the NBS data encryption standard,” *Computer, vol.* 10, pp. 74-84, 1977.

[50] A. DiPorto, “A public key cryptosystem based on a generalization of the knapsack problem,” presented at Eurocrypt ‘85, Linz, Austria, April 9-l 1, 1985.

[51] R. Eier and H. Lagger, “Trapdoors in knapsack cryptosystems,” in *Lecture Notes in Computer Science 149; Cryptography: Proc. Workshop Cryptography,* T. Beth, Ed., Burg Feuerstein, Germany, March 29-April 2, 1982, pp. 316-322. Berlin: Springer-Verlag, 1983.

[52] T. El Gamal, “A public key cryptosystem and a signature scheme based on discrete logarithms,” *IEEE Trans. Inform. Theory,* vol. IT-31, no. 4, pp. 469-472, July 1985.

[53] D. Estes, L. M. Adleman, K. Kompella, K. S. McCurley, and G. L. Miller, “Breaking the Ong-Schnorr-Shamir signature schemes for quadratic number fields,” in *Lecture Notes in Computer Science 218; Advances in Cryptology: Proc. Crypto* ‘85, H. C. Williams, Ed., Santa Barbara, CA, Aug. 18-22, 1985, pp. 3-13. Berlin: Springer-Verlag, 1986.

[54] A. M. Frieze, “On the Lagarias-Odlyzko algorithm for the subset sum problem,” *SIAM J. Comput.,* vol. 15, no. 2, pp. 536-539, May 1986.

[55] A. M. Frieze, J. Hastad, R. Kannan, J. C. Lagarias, and A. Shamir, “Reconstructing truncated integer variables satisfying linear congruences,” *SIAM J. Comput., vol.* **17,** pp. 262-280, 1988.

[56] A. M. Frieze, R. Kannan, and J. C. Lagarias, "Linear congruential generators do not produce random sequences," in *Proc. 25th IEEE Symp. Foundations of Computer Science*, Singer Island, FL, Oct. 24-26, 1984, pp. 480-484. Los Angeles: IEEE Computer Society Press, 1984.

[57] J. Gait, "Short cycling in the Kravitz-Reed public key encryption system," *Electron. Lett.*, vol. 18, pp. 706-707, 1982.

[58] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP—Completeness*, San Francisco: W. H. Freeman, 1979.

[59] J. von zur Gathen, D. Kozen, and S. Landau, "Functional decomposition of polynomials, '' in *Proc. 28th IEEE Symp. Foundations of Computer Science*, Los Angeles, CA, Oct. 12-14, 1987, pp. 127-131. Los Angeles: IEEE Computer Society Press, 1987.

[60] J. M. Goethals and C. Couvreur, "A cryptanalytic attack on the Lu-Lee public-key cryptosystem," *Philips J. Res.*, vol. 35, pp. 301-306, 1980.

[61] R. M. Goodman and A. J. McAuley, "A new trapdoor knapsack public key cryptosystem,'' in *Lecture Notes in Computer Science 209; Advances in Cryptology: Proc. Eurocrypt* '84, T. Beth, N. Cot, and I. Ingemarsson, Eds., Paris, France, April 9-11, 1984, pp. 150-158, Berlin: Springer-Verlag, 1985. Also in *IEE Proc.*, vol. 132, pt. E, no. 6, pp. 289-292, Nov. 1985.

[62] D. M. Gordon, "Discrete logarithms in $GF(p)$ using the number field sieve" (in press).

[63] J. A. Gordon, "Strong primes are easy to find," in *Lecture Notes in Computer Science 209; Advances in Cryptology: Proc. Eurocrypt* '84, T. Beth, N. Cot, and I. Ingemarsson, Eds., Paris, France, April 9-11, 1984, pp. 216-233. Berlin: Springer-Verlag, 1985.

[64] E. Grossman and B. Tuckerman, "Analysis of a Feistel-like cipher weakened by having no rotating key," IBM Research Rept. RC 6375, Jan. 31, 1977. Also in *Proc. ICC 78.*

[65] J. Hastad, personal communication.

[66] J. Hastad, B. Just, J. Lagarias, and C. I? Schnorr, "Polynomial time algorithms for finding integer relations among real numbers," *SIAM J. Comput., vol. 18,* pp. 859-881, 1989.

[67] J. Hastad and A. Shamir, "The cryptographic security of truncated linearly related variables," in *Proc. 17th ACM Symp. Theory of Computing*, Providence, RI, May 6-8, 1985, pp. 356-362. New York: Association for Computing Machinery, 1985.

[68] M. E. Hellman, "Another cryptanalytic attack on 'a cryptosystem for multiple communication'," *Inform. Processing Lett.*, vol. 12, pp. 182-183, 1981.

[69] M. E. Hellman, R. C. Merkle, R. Schroeppel, L. Washington, W. Diffie, S. Pohlig, and I? Schweitzer, "Results on an initial attempt to cryptanalyze the NBS data encryption standard," *Tech. Rept. SEL* 76-042, Stanford University, 1976.

[70] I? S. Henry, "Fast implementation of the knapsack cipher," *Bell Labs. Tech. J.*, vol. 60, pp. 767-773, May/June 1981.

[71] T. Herlestam, "Critical remarks on some public key cryptosystems," *BIT*, vol. 18, pp. 493-496, 1978.

[72] I? J. M. Hin, "Channel-error-correcting privacy cryptosystems," Ph.D. Thesis, Delft Univ. of Technology (1986, in Dutch).

[73] I. Ingemarsson, "A new algorithm for the solution of the knapsack problem," in *Lecture Notes in Computer Science 149; Cryptography: **Proc**. Workshop Cryptography,* T. Beth, Ed., Burg Feuerstein, Germany, March 29-April 2, 1982, pp. 309-315. Berlin: Springer-Verlag, 1983.

[74] N. S. James, R. Lidl, and H. Niederreiter, "Breaking the Cade cipher," in *Lec*ture Notes in *Computer Science* 263; Advances *in* Cryptology: ***Proc**. Crypto '86,* A. M. Odlyzko, Ed., Santa Barbara, CA, Aug. 11-15, 1986, pp. 60-63. Berlin: Springer-Verlag, 1987.

[75] R. R. Jueneman, "Analysis of certain aspects of output feedback mode," in *Advances in Cryptology: **Proc**. Crypto* '82, D. Chaum, R. L. Rivest, and A. T. Sherman, Eds., Santa Barbara, CA, Aug. 23-25, 1982, pp. 99-127. New York: Plenum Press, 1983.

[76] R. R. Jueneman, "Electronic document authentication," *IEEE Network, vol.* 1, no. 2, pp. 17-23, April 1987.

[77] B. S. Kaliski, R. L. Rivest, and A. T. Sherman, "Is the data encryption standard a group? (results of cycling experiments on DES)," *J. Cryptology,* vol. 1, no. 1, pp. 3-36, 1988.

[78] R. Kannan, "Improved algorithms for integer programming and related lattice problems," in ***Proc**. 15th ACM Symp. Theory **of** Computing,* pp. 193-206, 1983.

[79] D. E. Knuth, *Deciphering a Linear Congruential Encryption, Tech. Rept.* 024800, Stanford University, 1980, also in *IEEE Trans. Inform. Theory* vol. IT-31, pp. 49-52, Jan. 1985.

[80] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics **of** Computation,* vol. *48,* pp. 203-209, 1987.

[81] M. J. Kochanski, "Remarks on Lu and Lee's proposals," *Cryptologia,* vol. 4, no. 4, pp. 204-207, 1980.

[82] M. J. Kochanski, "A survey of data insecurity packages," *Cryptologia, vol.* 11, no. 1, pp. l-15, Jan. 1987.

[83] A. G. Konheim, *Cryptography,* A *Primer,* New York: John Wiley, 1981.

[84] D. Kravitz and I. Reed, "Extension of RSA cryptostructure: A Galois approach," *Electron. Lett., vol.* 18, pp. 255-256, 1982.

[85] H. Krawczyk, "How to predict congruential generators," in *Lecture Notes in Computer Science 435; Advances in Cryptology: **Proc**. Crypto* '89, G. Brassard, Ed., Santa Barbara, CA, Aug. 20-24, 1989, pp. 138-153. Berlin: Springer-Verlag, 1990.

[86] J. C. Lagarias, "Knapsack public key cryptosystems and diophantine approximation,"in *Advances in Cryptology: **Proc**. Crypto* '83, D. Chaum, Ed., Santa Barbara, CA, Aug. 22-24, 1983, pp. 3-23. New York: Plenum Press, 1984.

[87] J. C. Lagarias, "Performance analysis of Shamir's attack on the basic Merkle–Hellman knapsack cryptosystem," in *Lecture Notes in Computer Science 172; **Proc**. 11th Internat. Colloquium on Automata, Languages and Programming (ICALP),* J. Paredaens, Ed., pp. 312-323. Berlin: Springer-Verlag, 1984.

[88] J. C. Lagarias and A. M. Odlyzko, "Solving low density subset sum problems," *J. Assoc. Comp. Mach.,* vol. 32, pp. 229-246, 1985.

[89] J. C. Lagarias and J. Reeds, "Unique extrapolation of polynomial recurrences," *SIAM J. Comput.,* vol. 17, pp. 342-362, 1988.

[90] B. A. LaMacchia and A. M. Odlyzko, "Computation of discrete logarithms in prime fields," *Designs, Codes, and Cryptography,* vol. 1, pp. 46-62, 1991.

[91] P. J. Lee and E. F. Brickell, "An observation on the security of McEliece's public-key cryptosystem,'' in *Lecture Notes in Computer Science 330; Advances in Cryptology: Proc. Eurocrypt* '88, C. G. Gunther, Ed., Davos, Switzerland, May 25-27, 1988, pp. 275-280. Berlin: Springer-Verlag, 1988.

[92] A. K. Lenstra and H. W. Lenstra, Jr., "Algorithms in number theory," in *Handbook of Theoretical Computer Science,* J. Van Leeuwen, Ed., pp. 673-716. Cambridge, MA: MIT Press, 1990.

[93] A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovasz, "Factoring polynomials with rational coefficients," in *Math. Annalen 261,* pp. 515-534, 1982.

[94] A. K. Lenstra, "Factoring multivariate polynomials over finite fields," *J. Computer System Sci.,* vol. 30, no. 2, pp. 235-248, Apr. 1985.

[95] A. K. Lenstra, H. W. Lenstra, Jr., M. S. Manasse, and J. M. Pollard, "The number field sieve," in *Proc. 22nd ACM Symposium on Theory of Computing,* pp. 564-572, 1990.

[96] A. K. Lenstra and M. S. Manasse, "Factoring by electronic mail," in *Lecture Notes in Computer Science 434; Advances in Cryptology; Proc. Eurocrypt '89,* J.-J. Quisquater and J. Vandewalle, Eds., Houthalen, Belgium, April 10-23, 1989, pp. 355-37 1. Berlin: Springer-Verlag, 1990.

[97] H. W. Lenstra, Jr., "Integer programming with a fixed number of variables," *Math. Operations Res.,* vol. 8, no. 4, pp. 538-548, Nov. 1983.

[98] S. C. Lu and L. N. Lee, "A simple and effective public-key cryptosystem," *COMSAT Tech. Rev.,* pp. 15-24, 1979.

[99] F. Luccio and S. Mazzone, "A cryptosystem for multiple communication," *Inform. Proc. Lett.,* vol. 10, pp. 180-183, 1980.

[100] Y. Matyas and A. Shamir, "A video scrambling technique based on space filling curves," in *Lecture Notes in Computer Science 293; Advances in Cryptology: Proceedings* of *Crypto* '87, C. Pomerance, Ed., Santa Barbara, CA, Aug. 16-20, 1987, pp. 392-397. Berlin: Springer-Verlag, 1988.

[101] T. Matsumoto and H. Imai, "A class of asymmetric Crypto-systems based on polynomials over finite rings," in *Abstracts of Papers, IEEE Intern. Symp. Inform. Theory,* St. Jovite, Quebec, Canada, Sept. 26-30, 1983, pp. 131-132, 1983.

[102] K. S. McCurley, "A key distribution system equivalent to factoring," *J. Cryptology,* vol. 1, pp. 95-106, 1988.

[103] K. S. McCurley, "The discrete logarithm problem," in *Cryptography and Computational Number Theory (Proc. Symp. Appl. Math.),* C. Pomerance, Ed., pp. 49-74. Providence, RI: American Mathematics Society, 1990.

[104] R. J. McEliece, "A public-key cryptosystem based on algebraic coding theory," *DSN Progress Rept.* 42-44, Jet Propulsion Laboratory, pp. 114-116, 1978.

[105] R. C. Merkle and M. E. Hellman, "Hiding information and signatures in trapdoor knapsacks," *IEEE Trans. Inform. Theory, vol. 24,* no. 5, pp. 525-530, Sept. 1978.

[106] V. S. Miller, "Use of elliptic curves in cryptography," in *Lecture Notes in Computer Science 218; Advances in Cryptology: Proc. Crypto* '85, H. C. Williams, Ed., Santa Barbara, CA, Aug. 18-22, 1985, pp. 417-426. Berlin: Springer-Verlag, 1986.

[107] J. H. Moore, "Protocol failures in cryptosystems," this volume.

[108] J. H. Moore and G. J. Simmons, "Cycle structure of the DES for keys having palindromic (or antipalindromic) sequences of round keys," *IEEE Trans. Software Eng.*, vol. SE-13, no. 2, pp. 262-273, Feb. 1987.

[109] J. H. Moore and G. J. Simmons, "Cycle structure of the DES with weak and semiweak keys," in *Lecture Notes* in *Computer Science 263; Advances* in *Cryptology: Proc. Crypto* '86, A. M. Odlyzko, Ed., Santa Barbara, CA, Aug. 11-15, 1986, pp. 9-32. Berlin: Springer-Verlag, 1987.

[110] S. Murphy, "The cryptanalysis of FEAL-4 with twenty chosen plaintexts," *J. Cryptology,* vol. 2, no. 3, pp. 145-154, 1990.

[111] S. Miyaguchi, A. Shiraishi, and A. Shimizu, "Fast data encipherment algorithm FEAL-8," *Rev. Electrical Commun. Lab., vol. 36,* no. 4, pp. 433-437, July 1988.

[112] National Bureau of Standards, "Encryption algorithm for computer data protection," *Federal Register,* **vol. 40,** pp. **12134-12139,** March 17, 1975.

[113] National Bureau of Standards, "DES Modes of Operation," *Federal Information Processing Standard,* U.S. Department of Commerce, FIPS Pub. 81, Washington, DC, 1980.

[114] H. Niederreiter, "Knapsack-type cryptosystems and algebraic coding theory," in *Problems of Control and Information Theory, vol.* 15, no. 2, pp. 159-166, 1986.

[115] A. M. Odlyzko, "Cryptanalytic attacks on the multiplicative knapsack cryptosystern and on Shamir's fast signature system," *IEEE Trans. Inform. Theory,* vol. IT-30, no. 4, pp. 594-601, July 1984.

[116] A. M. Odlyzko, "Discrete logarithms in finite fields and their cryptographic significance," in *Lecture Notes in Computer Science 209; Advances in Cryptology: Proc. Eurocrypt* '84, T Beth, N. Cot, and I. Ingemarsson, Eds., Paris, France, April 9-11, 1984, pp. 224-314. Berlin: Springer-Verlag, 1985.

[117] T. Okamoto, "Fast public-key cryptosystems using congruent polynomial equations," *Electron. Lett.*, vol. 22, no. 11, pp. 581-582, 1986.

[118] T. Okamoto, "Modification of a public-key cryptosystem," *Electron. Lett.*, vol. *23,* no. 16, pp. 814-815, 1987.

[119] T. Okamoto and A. Shiraishi, "A fast signature scheme based on quadratic inequalities," in *Proc. IEEE Symp. Security and Privacy*, pp. 123-132, 1985.

[120] H. Ong and C. P. Schnorr, "Signatures through approximate representations by quadratic forms," in *Advances in Cryptology: Proc. Crypto* '83, D. Chaum, Ed., Santa Barbara, CA, Aug. 22-24, 1983, pp. 117-132. New York: Plenum Press, 1984.

[121] H. Ong, C. F! Schnorr, and A. Shamir, "An efficient signature scheme based on quadratic equations," in *Proc. 16th ACM Symp. Theory of Computing,* pp. 208-216, 1984.

[122] H. Ong, C. I? Schnorr, and A. Shamir, "Efficient signature schemes based on polynomial equations," in *Lecture Notes in Computer Science 196; Advances in Cryptology: Proc. Crypto* '84, G. R. Blakley and D. Chaum, Eds., Santa Barbara, CA, Aug. 19-22, 1984, pp. 37-46. Berlin: Springer-Verlag, 1985.

[123] M. Petit, *Etude mathématique de certains systèmes de chiffrement: Les sacs a dos,* (Mathematical study of some enciphering systems: The knapsack, in French), Ph.D. Thesis, Université de Rennes, France, 1982.

[124] J. P. Pieprzyk, "On public-key cryptosystems built using polynomial rings," in *Lecture Notes in Computer Science 219; Advances in Cryptology: Proc. Eurocrypt*

'85, F. Pilcher, Ed., Linz, Austria, April 1985, pp. 73-80. Berlin: Springer-Verlag, 1986.

[125] J. Boyar Plumstead, "Inferring a sequence generated by a linear congruence," in *Proc. 23rd IEEE Symp. Foundations Computer Science,* pp. 153-159, 1982.

[126] S. Pohlig and M. E. Hellman, "An improved algorithm for computing logarithms over GF(p) and its cryptographic significance," in *ZEEE Trans. Inform. Theory,* vol. IT-24, p. 106-110, 1978.

[127] J. M. Pollard and C. I? Schnorr, "An efficient solution of the congruence $x^2 + ky^2$ = m(mod *n)," IEEE Trans. Inform. Theory,* vol IT-33, no. 5, pp. 702-709, Sept. 1987.

[128] C. Pomerance, "Fast, rigorous factorization and discrete logarithm algorithms," in *Discrete Algorithms and Complexity,* D. S. Johnson, T. Nishizeki, A. Nozaki, and H. S. Wilf, Eds., New York: Academic Press, pp. 119-143, 1987.

[129] C. Pomerance, J. W. Smith, and R. Tuler, "A pipe-line architecture for factoring large integers with the quadratic sieve algorithm," *SIAM J. Comput.,* vol. 17, 387-403, 1988.

[130] J.-J. Quisquater and J. I? Delescaille, "How easy is collision search? Application to DES," in *Lecture Notes in Computer Science 434; Advances in Cryptology; Proc. Eurocrypt* '89, J.-J. Quisquater and J. Vandewalle, Eds., Houthalen, Belgium, April 10-23, 1989, pp. 429-435. Berlin: Springer-Verlag. 1990.

[131] M. Rabin, "Digitalized signatures," in *Foundations of Secure Computation,* R. A. DeMillo, D. I? Dobkin, A. K. Jones, and R. J. Lipton, Eds., pp. 155-168. New York: Academic Press, 1978.

[132] M. Rabin, "Digitalized signatures and public key functions as intractable as factorization," Massachusetts Institute of Technology, Laboratory for Computer Science, *Report MIT/LCS/TR-212,* Cambridge, MA, Jan. 1979.

[133] M. Rabin, "Probabilistic algorithms in finite fields," *SIAM J. Comput., vol. 9,* pp. 273-280, 1980.

[134] S. I? Radziszowski and D. L. Kreher, "Solving subset sum problems with the L^3 algorithm," *J. Combin. Math. Combin. Comput.,* vol. 3, pp. 49-63, 1988.

[135] T. R. N. Rao and K. H. Nam, "Private-key algebraic-coded cryptosystem," in *Lecture Notes in Computer Science 263; Advances in Cryptology: Proc. Crypto* '86, A. M. Odlyzko, Ed., Santa Barbara, CA, Aug. 11-15, 1986, pp. 35-48. Berlin: Springer-Verlag, 1987.

[136] J. A. Reeds and I? J. Weinberger, "File security and the UNIX system crypt command," *AT&T Bell Lab. Tech. J.,* vol. 63, no. 8, pp. 1673-1683, Oct. 1984.

[137] R. L. Rivest, L. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," in *Foundations of Secure Computation,* R. A. DeMillo, D. F! Dobkin, A. K. Jones, and R. J. Lipton, eds., New York: Academic Press, pp. 169-179, 1978.

[138] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Commun. ACM,* vol. 21, pp. 120-126, April 1978.

[139] R. A. Rueppel, "On the security of Schnorr's pseudo random generator," in *Lecture Notes in Computer Science 434; Advances in Cryptology; Proc. Eurocrypt* '89, J.-J. Quisquater and J. Vandewalle, Eds., Houthalen, Belgium, April 10-23, 1989, pp. 423-428. Berlin: Springer-Verlag, 1990.

[140] C. P. Schnorr, “A more efficient algorithm for a lattice basis reduction,” *J. Algorithms,* vol. 9, pp. 47-62, 1988.

[141] C. P. Schnorr, “A hierarchy of polynomial time lattice basis reduction algorithms,” *Theoretical Computer Sci.,* vol. 53, pp. 201-224, 1987.

[142] C. P. Schnorr, “On the construction of random number generators and random function generators,” in *Lecture Notes in Computer Science 330; Advances in Cryptology: Proc. Eurocrypt* ‘88, C. G. Giinther, Ed., Davos, Switzerland, May 25-27, 1988, pp. 225-232. Berlin: Springer-Verlag, 1988.

[143] I-? Schöbi and J. L. Massey, “Fast authentication in a trapdoor-knapsack public key cryptosystem,” in *Lecture Notes in Computer Science 149; Cryptography: Proc. Workshop Cryptography,* T. Beth, Ed., Burg Feuerstein, Germany, March 29-April 2, 1982, pp. 289-306. Berlin: Springer-Verlag, 1983.

[144] A. Shamir, “A fast signature scheme,” *MIT, Laboratory for Computer Science Report RM-107,* Cambridge, MA, July 1978.

[145] A. Shamir, “The strongest knapsack-based cryptosystem,” presented at Crypto *‘82.*

[146] A. Shamir, “A polynomial time algorithm for breaking the basic Merkle-Hellman cryptosystem,” in *IEEE Trans. Inform. Theory,* vol. IT-30, no. 5, pp. 699-704, Sept. 1984.

[147] A. Shamir, “On the security of DES,” in *Lecture Notes in Computer Science 218; Advances in Cryptology: Proc. Crypto* ‘85, H. C. Williams, Ed., Santa Barbara, CA, Aug. 18-22, 1985, pp. 280-281. Berlin: Springer-Verlag, 1986.

[148] A. Shamir, personal communication, Oct. 1985.

[149] A. Shamir, “The cryptographic security of compact knapsacks,” *MIT/LCS/TM-*164, Cambridge MA: Massachusetts Institute of Technology, 1980.

[150] A. Shamir, “On the cryptocomplexity of knapsack systems,” in *Proc. 11th ACM Symp. Theory Comput.,* pp. 118-129, 1979.

[151] A. Shamir and R. Zippel, “On the security of the Merkle-Hellman cryptographic scheme,” *IEEE Trans. Inform. Theory, vol. 26,* no. 3, pp. 339-340, May 1980.

[152] A. Shimizu and S. Miyaguchi, “Fast data encipherment algorithm FEAL,” in *Lecture* Notes *in Computer Science 304; Advances in Cryptology: Proc. Eurocrypt* ‘87, D. Chaum and W. L. Price, Eds., Amsterdam, The Netherlands, April 13-15, 1987, pp. 267-271. Berlin: Springer-Verlag, 1988.

[153] Z. Shmuely, “Composite Diffie-Hellman public-key generating systems are hard to break,” *Tech. Rept. no.* 356. Technion-Israel Institute of Technology Computer Science Department, Feb. 1985.

[154] G. J. Simmons, “A secure subliminal channel (?)’’ in *Lecture Notes in Computer* Science *218;* Advances *in Cryptology: Proc. Crypto* ‘85, H. C. Williams, Ed., Santa Barbara, CA, Aug. 18-22, 1985, pp. 33-41. Berlin: Springer-Verlag, 1986.

[155] J. Stern, “Secret linear congruential generators are not cryptographically secure,’’ in *Proc. 28th Symp. Foundations of Computer Science,* Los Angeles, CA, Oct. 12-14, 1987, pp. 421-426. Los Angeles: IEEE Computer Society Press, *1987.*

[156] R. Struik and J. van Tilburg, “The Rao-Nam scheme is insecure against a chosen-plaintext attack,” in *Lecture Notes in Computer Science 293; Advances in Cryptology: Proceedings of Crypto ‘87,* C. Pomerance, Ed., Santa Barbara, CA, Aug. 16-20, 1987, pp. 445-457. Berlin: Springer-Verlag, 1988.

[157] S. Tsujii, K. Kurosawa, T. Itoh, A. Fujioka, and T. Matsumoto, "A public-key cryptosystem based on the difficulty of solving a system of non-linear equations," *TSUJII Lab. Tech. Memorandum, no. 1,* 1986.

[158] B. Va!ee, M. Girault, and F! Toffin, "How to guess *l*-th roots modulo *n* when reducing lattice bases," in *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes (Proc. 6th Internat. Conf, AAECC-6),* T. Mora, Ed., Rome, Italy, July 4-8, 1988, pp. 427-442. Berlin: Springer-Verlag, 1989.

[159] B. Vallee, M. Girault, and F? Toffin, "How to break Okamoto's cryptosystem by reducing lattice bases," in *Lecture Notes in Computer Science 330; Advances in Cryptology: Proc. Eurocrypt* '88, C. G. Gunther, Ed., Davos, Switzerland, May 25-27, 1988, pp. 281-291. Berlin: Springer-Verlag, 1988.

[160] M. J. Wiener, "Cryptanalysis of short RSA secret exponents," *IEEE Trans. Inform. Theory, vol.* IT-36, pp. 553-558, 1990.

[161] M. Willett, "Trapdoor knapsacks without superincreasing structure," *Inform. Process. Lett.,* vol. 17, pp. 7-l 1, July 1983.

[162] H. C. Williams, "A modification of the RSA public-key encryption," in *IEEE Trans. Inform. Theory,* vol. IT-26, no. 6, pp. 726-729, 1980.

[163] H. C. Williams and B. Schmid, "Some remarks concerning the MIT public-key cryptosystem," *BIT,* vol. 19, pp. 525-538, 1979.

[164] M. Yagisawa, "A new method for realizing public-key cryptosystem," *Cryptologia,* vol. 9, no. 4, pp. 360-380, Oct. 1985.

[165] G. Yuval, "How to swindle Rabin," *Cryptologia,* vol. 3,, no. 3, pp. 187-190, July 1979.