Linearly Shift Knapsack Public-Key Cryptosystem

CHI-SUNG LAIH, JAU-YIEN LEE, SENIOR MEMBER, IEEE, LEIN HARN, MEMBER, IEEE, AND YAN-KUIN SU

***Abstract*—In this paper, we propose two algorithms to improve the Merkle–Hellman knapsack public-key cryptosystem. First, we propose an approach to transform a superincreasing sequence to a "high density" knapsack sequence. The algorithm is easy to implement and eliminates the redundancy of many knapsack cryptosystems. Second, a linearly shift method is used to improve the security of the knapsack public-key cryptosystem. We show that several knapsacks (e.g., the so-called "useless" knapsack), which cannot be generated by using the Merkle–Hellman scheme, can be generated by the linearly shift method. Thus, Shamir's attack to the original knapsack, as well as the low density attack to the iterated knapsack, cannot apply to our system successfully. It is interesting to note that the concept of the requirement of being one-to-one in practical enciphering keys is not necessary for our system.**

I. INTRODUCTION

DIFFIE and Hellman first proposed the idea of a public-key distribution system in 1976 [1]. However, the first implemented public-key cryptosystems were published by Rivest *et al.* [2] and Merkle *et al.* [3]. The security of both systems is based on the difficulty of factoring a large number and the complexity of knapsack problem, respectively. The first cryptanalysis to the basic Merkle–Hellman cryptosystem was published by Shamir [4]. Following his attack, some successful attacks to the iterated knapsack and the low density knapsack were proposed by Adleman [5] and Lagarias *et al.* [6]. Desmedt *et al.* [7] analyzed why these knapsack cryptosystems can be broken successfully. They showed that there exist some decodable enciphering keys but they cannot be obtained from Merkle–Hellman [3] or Graham–Shamir schemes [8]. Since these unobtainable keys are the worst cases in the knapsack problem, the exclusion of these keys from a cryptographic knapsack system explains the reason of success of several attacks on the knapsack algorithms. In [9] and [10], several general knapsack public-key cryptosystems were proposed to further reduce some of useless enciphering keys. They replaced the "easy" deciphering keys by random deciphering keys using linear algebra. However, many "useless" keys [7] still cannot be obtained by their schemes due to the constraint of modular multiplication. Besides, the data expansion is very large and the speed of decryption will be significantly slowed. Moreover, since they belong to the case of low density sequences, the low density attack can be applied to these systems. Other knapsack-type public-key cryptosystems have also been proposed. For example, Goodman *et al.* [12] proposed a scheme based on the modular transformation and Chinese Remainder Theorem, but it is recently shown that the system can be breakable [17]. Chor *et al.* [11] proposed a knapsack-type cryptosystem based on arithmetic in finite fields. However, the major problems for this scheme are the difficulty in the key generation and the slowness in the speed of decryption.

In this paper, a new knapsack-type public-key cryptosystem is proposed. The key generation is easy and cannot be obtained by applying one or more modular multiplications on any other sequence. It has been shown that the enciphering keys obtained from this algorithm have very high probability of falling into the category of the worst knapsack with NP-completeness. Therefore, Shamir's attack and the low density attack cannot be applied to our system.

This paper is organized as follows. In Section II we review the knapsack cryptosystem and the polynomial-time attack. Section III presents an algorithm to obtain the high density knapsack cryptosystem. A simple algorithm to improve the security of knapsack public-key cryptosystem is described in Section IV. In Section V, we show that even when the enciphering keys of the proposed system is not a one-to-one system, it can be used in cryptography. Conclusions are given in Section VI.

II. MERKLE–HELLMAN CRYPTOSYSTEM AND ITS POLYNOMIAL-TIME ATTACK

In the Merkle–Hellman cryptosystem, the receiver u_k first chooses a superincreasing sequence $\boldsymbol{B} = (b_1, b_2, \cdots, b_n)$ (i.e., $b_i > \sum_{j=1}^{i-1} b_j$), and then transfers $\boldsymbol{B}$ into a pseudorandom sequence $\boldsymbol{A} = (a_1, a_2, \cdots, a_n)$ by the following modulo transformation:

$$a_i = b_i * w \bmod M \tag{1a}$$

with

$$GCD(w, M) = 1 \tag{1b}$$

and

$$M > \sum_{i=1}^{n} b_i. \tag{1c}$$

Manuscript received August 1, 1988; revised January 15, 1989. This work was supported by the National Science Council, Republic of China, under Contract NSC 77-0404-E006-06. The material in this paper was presented in part at the 1988 IEEE International Symposium on Information Theory, Kobe, Japan, June 19–24, 1988.

C.-S. Laih, J.-Y. Lee, and Y.-K. Su are with the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan, Republic of China.

L. Harn is with the Computer Science Program, University of Missouri, Kansas City, MO 64110.

IEEE Log Number 8927179.

0733-8716/89/0500-0534$01.00

Finally, u_k publishes the numbers of $(a_1, a_2, \cdots, a_n)$ as the enciphering keys. On the transmitter side, the enciphering operation for a binary message $(x_1, x_2, \cdots, x_n)$ is given by

$$S = \sum_{i=1}^{n} x_i a_i. \tag{2}$$

Now, the transmitter sends S to the receiver through the insecure channel. Since $\boldsymbol{A}$ is public and S can be intercepted, an eavesdropper has to find a subset of $\boldsymbol{A}$ which sums up to S in order to obtain the message. This problem is known to be NP-complete. However, the intended receiver with the knowledge of $\boldsymbol{B}$ can obtain the message $(x_1, x_2, \cdots, x_n)$ by computing

$$S'' = S * w^{-1} \bmod M$$
$$= \sum_{i=1}^{n} b_i x_i \bmod M, \tag{3}$$

where $w^{-1} * w = 1 \bmod M$. It can easily be shown that the x_i can be found with at most n subtractions [3].

There are two major disadvantages in the Merkle–Hellman knapsack cryptosystem. First, the density is less than 1/2 [3] which is unfavorable to the transition efficiency and the size of the public file. This results from the superincreasing quality of $\boldsymbol{B}$ which causes a_i to be large and the corresponding density to be low [density $= n/\log_2 \max(a_i)$]. Second, because the deciphering keys are easy sequences, they are breakable [13], [5]–[7]. It has been proved [7] that there exist infinite pairs (v, M') which can transfer $\boldsymbol{A}$ to another superincreasing sequence $\boldsymbol{D}$. In order to find the pairs of (v, M'), conditions (1a), (1b), and (1c) can be reformulated to linear inequalities. Using the linear programming method proposed by Lenstra [14], it can be solved in polynomial time. Despite these two drawbacks, the Merkle–Hellman system has one major advantage. That is, the speed of enciphering and deciphering operations is faster than other well-known public keys (e.g., RSA). In the next two sections, we will show how to eliminate these two drawbacks in the Merkle–Hellman scheme.

III. High Density Knapsack Algorithm

In order to describe how to choose a superincreasing sequence which can transform to a "high density" knapsack sequence, let us prove the following theorem.

Theorem 1: If a superincreasing sequence b_i and an integer number v satisfy

$$b_i > \sum_{j=1}^{i-1} b_j + v, \qquad i = 1, 2, \cdots, n,$$
$$\text{and } M > \sum_{i=1}^{n} b_i \tag{4}$$

and for an integer c_i such that $0 \le c_i \le v$, then $b_i' = b_i - c_i$ still forms a superincreasing sequence with $M > \Sigma_{i=1}^{n} b_i'$.

Proof: Since $b_i' = b_i - c_i$ and $0 \le c_i \le v$, we have

$$\sum_{j=1}^{i-1} b_j' = \sum_{j=1}^{i-1} b_j - \sum_{j=1}^{i-1} c_j \le \sum_{j=1}^{i-1} b_j.$$

Since $b_i > \Sigma_{j=1}^{i-1} b_j + v$, we have

$$b_i' = b_i - c_i \ge \sum_{j=1}^{i-1} b_j + v - c_i \ge \sum_{j=1}^{i-1} b_j \ge \sum_{j=1}^{i-1} b_j'$$

and $M > \Sigma_{i=1}^{n} b_i \ge \Sigma_{i=1}^{n} b_i'$. Q.E.D.

Although c_i is limited in $\{0 \le c_i \le v\}$, through the transformation, $(*w \bmod M)$, it can be arbitrary chosen and distributed in $[0, vw]$ to reduce the enciphering keys for high density sequences. Now, let us describe the procedures for generating this high density sequence.

Step 1: Randomly choose a superincreasing sequence $\boldsymbol{B} = (b_1, b_2, \cdots, b_n)$ and two integers w, M satisfying $\mathrm{GCD}(w, M) = 1$ and (4), where $v = \lfloor M/w \rfloor$ (where $\lfloor x \rfloor$ is a floor function, representing the largest integer value smaller than x).

Step 2: Calculate the original enciphering keys $a_i = b_i * w \bmod M$, then $a_i < M$ for all i. (5)

Step 3: Compute and public the high density enciphering keys, $a_i' = a_i \bmod w$, then $a_i' < w$ for all i. (6a)

Step 4: Calculate $c_i = \lfloor a_i/w \rfloor$, then $0 \le c_i \le v$, and compute the deciphering keys $b_i' = b_i - c_i$. (6b)

It can be easily proved that $a_i' = b_i' * w \bmod M$ and $a_i' < w$. Using theorem 1, we can show that b_i' is a superincreasing sequence and satisfies $M > \Sigma_{i=1}^{n} b_i'$. It is obvious that the original Merkle–Hellman enciphering keys distributed in $[1, M]$ have been reduced to a_i' distributed in $[1, w]$ but with the same security. The density can be controlled by properly choosing w which is much less than M.

Example 1: If $n = 6$, $m = 8443$, and $w = 259$, where M, w satisfy (1b), then $v = \lfloor M/w \rfloor = 32$.

Step 1: Randomly choose a superincreasing sequence $B = (111, 189, 445, 770, 2399, 4325)$ satisfying (4).

Step 2: Calculate $A = (3420, 6734, 5496, 5241, 5002, 5699)$.

Step 3: Calculate $A' = (53, 2, 57, 61, 81, 1)$.

Step 4: Calculate $C = (13, 26, 21, 20, 19, 22)$ and $B' = (98, 163, 424, 750, 2380, 4303)$.

It is easy to check that $a_i' = b_i' * w \bmod M$.

As Merkle and Hellman suggested, if $n = 100$, b_i is within the range of $[2^{n+i-1} - 2^n + 1, 2^{n+i} - 1]$, M is chosen uniformly from the numbers between 2^{201} and $2^{202} - 1$. We suggest that w is within the range of $[2^{105} + 1, 2^{106}]$. The density of our proposed algorithm is higher than 0.94 in comparison to 0.5 obtained by the original Merkle–Hellman scheme. In general, many existing knapsack cryptosystems such as the Graham–Shamir system can be improved by our scheme.

IV. Linearly Shift Knapsack Cryptosystem

The high density knapsack system proposed in Section III, however, is a special case of the Merkle–Hellman scheme. Therefore, it is breakable by Shamir's attack [12] or other attacks proposed by Brickell [15], Adleman [5], and Lagarias *et al.* [6]. In this section, we introduce a linearly shift method to help us to generate enciphering keys which cannot be obtained through single or mul-

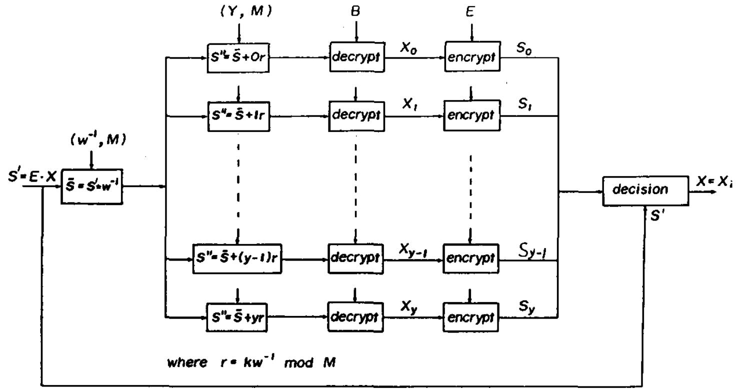


Fig. 1. The description of parallel decryption architecture for the linearly shift knapsack cryptosystem.

tiple multiplications. As a result, the similar cryptanalytic techniques mentioned above cannot crack our system.

We describe the linearly shift knapsack cryptosystem as follows.

Step 1: Randomly choose an easy knapsack sequence $\boldsymbol{B} = (b_1, b_2, \cdots, b_n)$.

Step 2: Transfer this easy knapsack sequence into a hard knapsack sequence $\boldsymbol{A}$ by modular multiplications using (1).

Step 3: Choose a random binary sequence $\boldsymbol{Q} = (q_1, q_2, \cdots, q_n)$ and an integer k with $0 < k < \min \cdot (a_i)$ for $q_i = 1$. Then a_i are linearly shifted by performing $e_i = a_i - kq_i$ and e_i are published as the public enciphering keys. The deciphering keys for intended receivers are $(\boldsymbol{B}, k, w, M)$.

If the receiver receives $S = \Sigma_{i=1}^{n} a_i x_i$, where $\boldsymbol{X} = (x_1, x_2, \cdots, x_n)$ is the message, he can decipher S properly just by following the normal decryption procedure [3]. However, the receiver receives $S' = \Sigma_{i=1}^{n} e_i x_i$ instead of S. From step 3 mentioned above, we obtain

$$\begin{aligned} S * w^{-1} &= \left(\sum_{i=1}^{n} a_i x_i \right) * w^{-1} \mod M \\ &= \sum_{i=1}^{n} (e_i + kq_i) x_i * w^{-1} \mod M \\ &= S' * w^{-1} + r * \sum_{i=1}^{n} q_i x_i \mod M \end{aligned} \tag{7}$$

where $r = kw^{-1} \bmod M$.

Since $\boldsymbol{Q}$ and $\boldsymbol{X}$ are binary sequences, which implies $0 \leq \Sigma_{i=1}^{n} q_i x_i \leq y \leq n$, where $y = \Sigma_{i=1}^{n} q_i$. Thus, the receiver can guess the correct $S * w^{-1} \bmod M$ at most $y + 1 \leq n + 1$ times. If the system is one-to-one, the rightness of guessing can be easily verified through normal enciphering procedures. According to Shamir's theorem [16], "a random modular knapsack system with n generators and modular M is likely to be one-to-one when $n < (\log_2 M)/2$ and non-one-to-one otherwise." That is, from this equation if M is chosen larger than 2^{2n}, then the system is likely to be one-to-one. The parallel decryption procedure is shown in Fig. 1. As shown in Fig. 1, the ciphertext S' is first transformed by the secret key pair (w^{-1}, M) to $\bar{S}$. Since the enciphering keys are shifted, as shown in (7), the receiver adds $j * r, j = 0, 1, \cdots, y$ to the S and decrypts the messages $\boldsymbol{X}_j, j = 0, 1, \cdots, y$ by using the superincreasing sequence $\boldsymbol{B}$. These messages $\boldsymbol{X}_j$, $j = 0, 1, \cdots, y$ contain the corrected message $\boldsymbol{X}$, but the receiver does not know which is the corrected one. However, through the encryption procedure and in comparison to the original ciphertext S', it is easy to find the corrected message $\boldsymbol{X}$. As shown in Fig. 1, the complexity of decryption is about 1 multiplication, n subtractions, and $n + 1$ additions.

Remarks: For a better uncertainty, the authors suggest that $\boldsymbol{Q}$ can be one of the following two types.

a) $\boldsymbol{Q}$ is an arbitrary binary sequence with $\Sigma_{i=1}^{n} q_i = y = n/2$.

b) $\boldsymbol{Q} = (q_1, q_2, \cdots, q_n)$ with $q_i \in \{-1, 0, 1\}$ and $\Sigma_{i=1}^{n} q_i = 0$.

In general, if $\boldsymbol{Q}$ is m-ary, the paths shown in Fig. 1 must be $(m - 1)y + 1$. However, the speed of decryption remains unchanged.

Now, we prove that the inverse transform of $\boldsymbol{E} = (e_1, e_2, \cdots, e_n)$ by (w^{-1}, M) does not form an easy knapsack sequence.

$$\begin{aligned} e_i * w^{-1} &= (a_i - kq_i) * w^{-1} \mod M \\ &= a_i * w^{-1} - kq_i * w^{-1} \mod M \\ &= \begin{cases} M + b_i - r & \text{for } q_i = 1 \text{ and } b_i < r \\ b_i - r & \text{for } q_i = 1 \text{ and } b_i \geq r \\ b_i & \text{for } q_i = 0 \end{cases} \end{aligned} \tag{8}$$

where $r = k * w^{-1} \bmod M$.

Obviously, the cryptographer can control r such that the inverse transformation is not an easy sequence and also does not satisfy (1c).

In fact, if $\boldsymbol{E}$ is mapped from another easy sequence or random sequence satisfying (1c), then the security of this algorithm is equal to the Merkle–Hellman system or the systems proposed in [8]–[10]. In [15], it is shown that the probability for a random sequence to be an image of a superincreasing sequence under modular transformation [i.e., (1)] is less than $2^{-\binom{n}{2}} * (\Sigma_{i=1}^n e_i)^2$. Now, we use the following theorem to prove that the probability for a random sequence to be an image of another random sequence is very small.

Theorem 2: Assume that $\boldsymbol{E} = (e_1, e_2, \cdots, e_n)$ are uniformly distributed, independent random variables in $[1, M]$. The probability P that $\boldsymbol{E}$ be the image of a sequence $\boldsymbol{H}$, under a modular transformation (1c), satisfies

$$P < \left(\sum_{i=1}^{n} e_i\right)/n! < nM/n! \tag{9}$$

Proof: It can be shown that if we randomly choose n integer h_i from $[1, M]$, the probability p satisfying $\Sigma_{i=1}^n h_i < M$ is $p < 1/n!$.

Since there are at most $\Sigma_{i=1}^n e_i$ minima divide to $\Sigma_{i=1}^n e_i$ interval for the function $g_i(t) = e_i t - s_i M$ (see [15]), where $s_i = \lfloor e_i t/M \rfloor$. Hence, if we "test" every point in $[1, M]$, we must find a modular transform (w_j, M), $1 \le j \le \Sigma_{i=1}^n e_i^{-1}$, if it exists. The probability, P, of success (assuming the independence of $g_i(t)$ values at the test points) is

$$P < \left(\sum_{i=1}^{n} e_i\right)/n! < nM/n! \qquad \text{Q.E.D.}$$

According to [15] and Theorem 2, we know that the probability for enciphering keys e_i generated by our algorithm being the image of a superincreasing sequence is less than 2^{-4536} and being the image of random sequence is less than 10^{-95} when $n = 100$ and $M = 2^{200}$. In other words, $\boldsymbol{E}$ have very large probability falling into the worst case of the knapsack problem. For a cryptanalyst, however, it is an NP problem unless he can guess k and $\boldsymbol{Q}$.

Example 2:

Step 1: As shown in Example 1, we choose $B' = [98, 163, 424, 750, 2380, 4303]$ with $M = 8443$, $w = 259$.

Step 2: Transfer B' into a hard sequence $A' = [53, 2, 57, 61, 81, 1]$.

Step 3: Randomly choose $Q = [1, 0, 1, 1, 1, 0]$ and $k = 42$, then the enciphering keys $E = [11, 2, 15, 19, 39, 1]$.

In this example, $\boldsymbol{E}$ is found with worst case property which cannot be obtained from other key generation algorithms [7]. In general, this algorithm could be used to improve almost all knapsack cryptosystems such as [8]–[10]. It is easy to see that those systems are a special case of our system with $q_i = 0$ for all i.

V. Non-One-to-One System Behavior in Our Knapsack Cryptosystem

An interesting result of the linearly shift method described in the above section is that if the receiver publishes $\boldsymbol{Q}$, then the enciphering keys can still be used in a knapsack cryptosystem even if these keys are non-one-to-one. The key point is that when $\Sigma_{i=1}^n x_i q_i$ is known, the receiver can obtain $S = \Sigma_{i=1}^n x_i a_i = \Sigma_{i=1}^n e_i x_i + k \Sigma_{i=1}^n q_i x_i$. Since $\boldsymbol{A}$ is a one-to-one system (which is not published), the receiver can decrypt $\boldsymbol{X}$ uniquely. Before we describe how the system works, let us prove the following theorem.

Theorem 3: Let $\boldsymbol{A} = (a_1, a_2, \cdots, a_n)$ be a one-to-one system and $e_i = a_i - kq_i$, $0 < k < \min \cdot (a_i)$ for $q_i = 1$. If $\Sigma_{i=1}^n x_i e_i = \Sigma_{i=1}^n y_i e_i$ where $X \neq Y$, then $\Sigma_{i=1}^n x_i q_i \neq \Sigma_{i=1}^n y_i q_i$.

Proof: Let $\Sigma_{i=1}^n x_i q_i = t_1$, $\Sigma_{i=1}^n y_i q_i = t_2$ and $X \neq Y$. Since $\Sigma_{i=1}^n x_i e_i = \Sigma_{i=1}^n y_i e_i$. Thus, $\Sigma_{i=1}^n x_i e_i + kt_1 = \Sigma_{i=1}^n y_i e_i + kt_2 + k(t_1 - t_2)$. Then $\Sigma_{i=1}^n a_i x_i = \Sigma_{i=1}^n y_i a_i + k(t_1 - t_2)$. Since $\boldsymbol{A}$ is a one-to-one system, it implies that

$$\sum_{i=1}^{n} a_i x_i \neq \sum_{i=1}^{n} a_i y_i,$$

$$\therefore t_1 \neq t_2. \qquad \text{Q.E.D.}$$

From theorem 3, we see that even if $\boldsymbol{E}$ is a non-one-to-one system, the receiver can decrypt $\boldsymbol{X}$ uniquely if he knows $\Sigma_{i=1}^n x_i q_i$. For convenience, we assume that $q_i = 1$ for $i = 1, 2, \cdots, y$ and $q_i = 0$ for $i = y + 1, y + 2, \cdots, n$ (if $\boldsymbol{Q}$ is not of this form, it can be obtained by scrambling). y must be smaller than $\lfloor n - \log_2 n \rfloor = t$. Now, the transmitter divides the binary message into many blocks. Each block contains t bits for the message and $n - t = m$ bits for the information $\Sigma_{i=1}^y q_i x_i$. Assume the ith block of message is $\boldsymbol{X}_i = (x_{i1}, x_{i2}, \cdots, x_{in})$. The transmitter computes

$$\sum_{j=1}^{y} x_{ij} q_j = Z_i = (z_{i,m}, z_{i,m-1}, \cdots, z_{i,1})$$

where $z_{i,j} = 0$ or 1; $j = 1, 2, \cdots, m$.

The transmitter then puts Z_i into the $(i - 1)$th block, that is,

$$\begin{aligned} x_{i-1,n-m+1} &= z_{i,m} \\ x_{i-1,n-m+2} &= z_{i,m-1} \\ &\vdots \\ x_{i-1,n} &= z_{i,1} \end{aligned} \tag{10}$$

for $1 \le i \le u$, where u is the number of blocks.

For the first block, $x_{0,j} = 0$, $0 \le j \le y$, and $x_{0,j}$ is a randomly chosen binary bit for $y < j \le t$. The transmitter can encrypt each block message into ciphertext using enciphering keys. When the receiver obtains the 0th block ciphertext, he can obtain the message uniquely (since $S_0' = S_0 = \Sigma_{i=1}^n a_i x_{0,i} = \Sigma_{i=1}^n e_i x_{0,i}$ in the 0th block). Since the next block's information $\Sigma_{i=1}^n x_i q_i$ is connected to the $(i - 1)$th block message, the receiver can decrypt the ith message uniquely. Thus, if $\boldsymbol{Q}$ is public, $\boldsymbol{E}$ still can be used in the cryptosystem even it is non-one-to-one. Under this condition, the high density algorithm proposed in Section III can be used to reduce the ratio of data expansion and the public file memory.

VI. Conclusion

This paper proposes two algorithms to eliminate the drawbacks of the knapsack public-key cryptosystem: a high density algorithm which can reduce the data expansion ratio and the size of the public file to about one-half of the Merkle-Hellman knapsack cryptosystem, and the linearly shift knapsack algorithm which can improve the system security. It was shown that the enciphering keys generated by the linearly shift knapsack algorithm have a very large probability [about $1 - (nM/n!)$] of falling into the worst case of the knapsack problem. In Section V, we also showed that even when the enciphering keys are non-one-to-one, it still can be used in cryptography.

The ideal knapsack cryptosystem would have two characteristics: 1) the density is close to 1; 2) the enciphering keys cannot be transformed by single or multiple multiplications. Note, if the high density algorithm is used first, it cannot guarantee that the enciphering keys generated by the linearly shift algorithm are one-to-one; although publishing Q and using the protocol proposed by Section V can overcome this problem. However, it may reduce the system security. Therefore, further work is seen to be needed in this area. Is there an algorithm to choose k and Q which guarantees the enciphering keys belong to one-to-one when high density algorithm is used first? If the answer is positive, then the high density algorithm and linearly shift knapsack algorithm may construct an ideal knapsack cryptosystem.

The knapsack public-key cryptosystems have the main advantage that the speed of both encryption and decryption are much faster than other well-known public-key cryptosystems. For example, RSA needs about $3n/2$ modular multiplications for both encryption and decryption, while in our system it needs only about n additions for encryption and 1 multiplication, n subtractions, and $n + 1$ additions for decryption. Therefore, when the linearly shift knapsack cryptosystem is acceptable in secure cryptography, the high throughput of the system and ease of implementation will make it an attractive alternative.

References

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

[2] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystem," *Commun. ACM*, vol. 21, no. 2, pp. 120-126, Feb. 1978.

[3] R. Merkle and M. E. Hellman, "Hiding information and signatures in trapdoor knapsack," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 525-530, Sept. 1978.

[4] A. Shamir, "A polynomial time algorithm for breaking the basic Merkle-Hellman cryptosystem," in *Proc. 23rd Annu. Symp. Foundations of Comput. Sci.*, Nov. 1982, pp. 145-152. Also *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 699-704, Sept. 1984.

[5] L. Adleman, "On breaking generalized knapsack public key cryptosystems," Univ. Southern Calif., Los Angeles, Internal Rep. TR-83-207, Mar. 1983.

[6] J. C. Lagarias and A. M. Odlyzko, "Solving low-density subset sum problems," in *Proc. 24th Annu. Symp. Foundations Comput. Sci.*, 1983, pp. 1-10. Also *J. Assoc. Comput. Machine*, vol. 32, pp. 229-246, 1985.

[7] Y. G. Desmedt, J. P. Vandewalle, and R. M. Govarets, "A critical analysis of the security of knapsack public-key algorithms," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 601-611, July 1984. Also presented at IEEE Int. Symp. Inform. Theory, Les Arcs, France (Abstract of papers), June 1982, pp. 115-116.

[8] A. Shamir and R. E. Zippel, "On the security of the Merkle-Hellman cryptographic scheme," *IEEE Trans. Inform. Theory*, vol. IT-26, pp. 339-340, May 1980.

[9] Y. G. Desmedt, J. P. Vandewalle, and R. M. Govarets, "A general public key cryptographic knapsack algorithm based on linear algebra," in *Proc. IEEE Int. Symp. Inform. Theory* (Abstract of papers), st. Jovite, P.Q., Sept. 26-30, 1983, pp. 129-130.

[10] A. Shamir, "Embedding cryptographic trapdoors in arbitrary knapsack systems," *Inform. Processing Lett.*, no. 17, pp. 77-79, Aug. 1983.

[11] B. Chor and R. L. Rivest, "A knapsack type public key cryptosystem based on arithmetic in finite fields," in *Advances in Cryptology: Proceedings of Crypto '84.* Berlin: Springer-Verlag, 1985, pp. 54-65. (A revised version to appear in *IEEE Trans. Inform. Theory*.)

[12] R. M. F. Goodman and A. J. McAuley, "A new trapdoor knapsack public-key cryptosystem," in *Advances in Cryptology: Proceedings of Eurocrypto '84.* Berlin: Springer-Verlag, 1985, pp. 150-158. Also *IEE Proceedings*, vol. 132, pt. E, no. 6, pp. 289-292, Nov. 1985.

[13] E. F. Brickell, "Solving low density knapsacks in polynomial time," in *Proc. IEEE Int. Symp. Inform. Theory* (Abstract of papers), St. Jovite, P.Q., Canada, Sept. 26-30, 1983, p. 130.

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

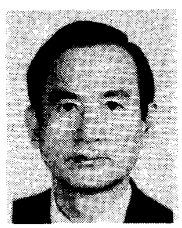
[15] E. F. Brickell and G. J. Simmons, "A status report on knapsack based public key cryptosystems," Sandia Nat. Lab. Rep., 1983.

[16] A. Shamir, "On the cryptocomplexity of knapsack systems," in *Proc. Symp. ACM Theory Comput.*, vol. 11, 1979, pp. 118-129.

[17] E. F. Brickell and A. M. Odlyzko, "Cryptanalysis: A survey of recent results," *Proc. IEEE*, vol. 76, pp. 578-593, May 1988.

Chi-Sung Laih was born in Cha-I, Taiwan, on June 4, 1956. He received the B.S. and M.S. degrees in electrical engineering from National Cheng Kung University, Tainan, Taiwan, Republic of China, in 1984 and 1986, respectively.

Since 1986 he has been an Instructor in the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan. He is presently a Ph.D. candidate at the Institute of Electrical Engineering, National Cheng Kung University. His research interests include cryptography, coding theory, and communications.



Jau-Yien Lee (M'84–SM'88) was born in Fukien, China, on August 7, 1928. He received the B.S., M.S., and Ph.D. degrees in electrical engineering from National Cheng Kung University, Tainan, Taiwan, in 1966, 1970, and 1974, respectively.

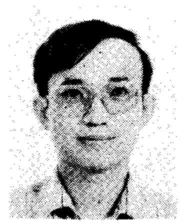
From 1977 to 1978 he was a Postdoctorate at San Jose State University, CA. From 1948 to 1961 he served as an Instructor in the Army Signal School where he taught radio electronics. From 1970 to 1982, upon transferring to the Military Academy, he served as a Faculty member and later as Chairman of the Electrical Engineering Department of the Academy. From 1978 to 1982 he was also an Adjunct Professor in the Department of Electrical Engineering at National Cheng Kung University. In 1982 he became Professor of Electrical Engineering, and he is now Chairman of the Department of Electrical Engineering. His principal fields of research are the areas of CAD/VLSI, signal processing, communications, and thin film circuitry.

Dr. Lee is a member of AMSE, CIEE, and Phi Tau Phi.

Lein Harn (M'85) was born in Taipei, Taiwan, in 1954. He received the B.S. degree from the National Taiwan University in 1977, the M.S. degree from the State University of New York, Stony Brook, in 1980, and the Ph.D. degree from the University of Minnesota, Minneapolis, in 1984, all in electrical engineering.

From 1981 to 1984 he was a Research/Teaching Assistant and was involved in research on signal detection and digital filtering in the Department of Electrical Engineering at the University of Minnesota. Since 1984 he has been an Assistant Professor in the Department of Electrical and Computer Engineering, University of Missouri at Columbia, and in the Department of Computer Science, University of Missouri at Kansas City. From 1986 to 1987 he was a Visiting Associate Professor at the National Cheng Kung University, Taiwan, R.O.C. Currently he is a full-time Faculty member at the Computer Science Department, University of Missouri at Kansas City. His research interests include digital filter design, multidimensional digital signal processing, image processing, system performance evaluation and modeling, 3-D image processing, and cryptography.

Dr. Harn is a member of Eta Kappa Nu.



Yan-Kuin Su was born in Kaohsiung, Taiwan, on August 23, 1948. He received the B.S. and Ph.D. degrees in electrical engineering from National Cheng Kung University, Taiwan, in 1971 and 1977, respectively.

From 1977 to 1983 he was with the Department of Electrical Engineering, National Cheng Kung University, Taiwan, as an Associate Professor, and was engaged in research on compound semiconductors and optoelectronic materials. In 1983 he was promoted to Professor of Electrical Engineering.

From 1979 to 1980 and 1986 to 1987 he was on leave and working at the University of Southern California and AT&T Bell Laboratories as a Visiting Scholar, respectively. He has published over 100 papers in the area of thin film materials and devices and optoelectronic devices. His current interests include compound semiconductors, integrated optics, and microwave devices.

Dr. Su is a member of the Chinese Society of Engineering and Phi Tau Phi.