通信情報保護學會論文誌
第 1 卷, 第 1 號, 1991. 12

Knapsack 공개키 암호법의 효율적인 구현

김세헌* · 엄봉식*

An Efficient Construction of Chor-Rivest Knapsack Cryptosystem

Se-hun Kim and Bong-sik Um

요 약

Knapsack 암호체계는 NP-Complete인 knapsack 문제에 기초한 공개키 암호체계이다. 이러한 암호체계의 안전성에 관하여서는 그동안 많은 논란이 있어 왔다. 쉬운 knapsack 문제를 모듈라 연산으로 숨기는 거의 모든 knapsack 암호체계는 lattice basis reduction 알고리즘으로 암호분석된다. 그러나 유한체에 근거한 새로운 종류의 knapsack 암호체계가 계속하여 개발되어 왔다. 특히 Bose-Chowla 정리에 근거하여 모듈라 연산을 사용하지 않는 Chor-Rivest knapsack 암호체계는 기존의 모든 암호분석 방법에 대하여 안전한 것으로 알려져 있다.

본 연구에서는 knapsack 문제를 정수계획법 문제로 변환하고 이를 이완하여 해를 구함으로써 knapsack 문제의 부분해를 구할 수 있음을 보인다. 이는 일반적인 knapsack 암호체계에 대한 효율적인 부분 암호분석 방법이 된다. 또한 안전한 것으로 알려진 Chor-Rivest 암호체계는 구현상의 효율성이 크게 문제 되고 있다. 일반화된 Bose-Chowla 정리에 근거하여, 구현상의 효율성이 제고된 안전한 knapsack 공개키 암호체계를 제시하고자 한다.

Abstract

Knapsack public-key cryptosystems are based on the knapsack problem which is NP-complete. All of the knapsack cryptosystems which use some form of modular multiplication to disguise an easy embedded knapsack problem, are known to be insecure. However, the Chor and Rivest knapsack cryptosystem based on arithmetic in a finite field is secure against all known cryptanalytic attacks.

We suggest a new method of attack on knapsack cryptosystem which is based on the relaxation

* 한국과학기술원

of a quadratic 0-1 integer optimization problem. We show that under certain conditions some bits of the solution of knapsack problem can be determined by using persistency property of linear relaxation.

Also we propose a new Chor-Rivest cryptosystem which is based on an extension of Bose-Chowla theorem. Unlike the original Chor-Rivest system, this new cryptosystem reduces the number of calculations of discrete logarithms which are necessary for the implementation in a multi-user system.

1. 서 론

1976년 공개키 암호체계의 개념이 제시된 이후로 이를 처음 구현한 것이 Merle-Hellman 암호체계와 RSA 암호체계이다. 그 이후로 많은 공개키 암호체계가 개발되었으며, 이들은 대개 이산로그와 같은 어려운 정수이론 문제에 근거하고 있거나, knapsack 문제에 기초하고 있다.

Knapsack 공개키 암호체계는 knapsack 문제에 trapdoor를 부가함으로써 구축한 공개키 암호로서 Merkle과 Hellman[16]이 처음으로 이를 구축한 이후, 이를 암호분석하고자 하는 많은 시도들이 있어 왔으며 이들은 knapsack 암호의 분석에 성공하고 있다[10]. 이 결과, 이들 knapsack 공개키 암호체계는 그 안전성에 많은 문제가 있어 이를 실제로 구축하는 것은 문제가 있다. 그러나 암호화 및 복호화 과정이 간편하다는 등의 장점을 지니고 있어서 많은 연구의 대상이 되어 왔으며, 새로운 knapsack 암호체계가 계속하여 개발되어 왔다.

현재까지의 대부분의 knapsack 암호는 쉬운 knapsack 벡터를 모듈라 변환시킴으로써 trapdoor knapsack을 구성하고 있다. 이러한 종류의 knapsack 암호는 Lovasz[14]의 lattice basis reduction 알고리즘을 이용하여 개별적인 구조적 특성을 밝혀내려는 접근방법이나 일반적인 knapsack 문제의 해를 구하려는 접근방법에 의해 모두 암호분석이 되었다. Chor와 Rivest[7]의 knapsack 암호는 유한체에서의 연산에 근거한 새로운 knapsack 암호이며 이는 아직까지 기존의 암호분석에 대하여 안전하다.

본 논문에서는 대표적인 knapsack 암호를 살펴보고, Lagarias와 Odlyzko[12]의 저밀도 암호분석을 중심으로 하여 기존의 knapsack 암호에 대한 암호분석을 고찰하였다. 또한 일반적인 knapsack 문제를 비제약 0−1 정수 계획법으로 변환시키고 이를 roof 함수로 이완시킨 문제의 해와 비제약 0−1 정수계획법의 해를 고찰함으로써, knapsack 문제의 부분해를 구할 수 있음을 밝히고자 한다. 또한 안전하지만 구현상의 효율성에 문제가 있는 Chor-Rivest 암호체계를 일반화된 Bose-Chowla 정리에 근거하여 개선하고자 한다.

2. Knapsack 암호체계

Knapsack 문제는 일정한 비음정수의 가중치 벡터 $a=(a_1, a_2, \cdots, a_k)$와 비음정수 S가 주어졌을 때, S가 a의 부분합이 되는지 또한 그렇다면 a의 어떤 원소들의 합이 S가 되는지 알아내는 문제이다. 즉 $S=ax=\Sigma a_ix_i$를 만족하는 0−1 벡터 x를 찾아내는 문제이다. 정수 벡터 a에서 그 부분집합의 합 S를 구하는 것은 쉽지만, 그 역과정인 knapsack 문제는 Combinatorics에서 잘 알려진 문제로 일반적으로 풀기 어려운 NP-complete 문제이다. 가장 효율적인 Schroeppel[22]의 알고리즘으로도 knapsack 문제를 풀기 위해서는 $0(2^{n/2})$의 시간과 $0(2^{n/4})$의 기억장소를 필요로 한다. 전형적인 knapsack 공개키 암호체계는 가중치 벡터 a를 공개 암호키로 공개하고 0−1 벡터인 메시지는 ax로 암호화된다. 이 knapsack 문제의 해를 구하는 것이 knapsack 암호를 복호화시키는 것이 된다. 복호화 과정이 가능케

하기 위해 knapsack 벡터에 일정한 비밀정보를 부가하여 이를 이용하면 복호화가 가능하도록 해준다.

knapsack 문제는 일반적으로 풀기 어려운 문제이지만 모든 knapsack 문제가 그러한 것은 아니며, 특수한 구조를 갖는 많은 경우에 knapsack 문제의 해를 구하기가 용이하다. Merkle과 Hellman[16]은 초증가하는 정수 벡터 a를 구성하고 이것에 모듈라 연산을 함으로써 복잡한 knapsack 벡터로 전환시켰다. 즉 knapsack 일방함수에 trapdoor를 부가하여 trapdoor knapsack 공개키 암호체계를 개발하였다.

Knapsack 벡터 a에서 각 원소의 크기가 그 이전의 원소의 합보다 큰 경우에 a는 초증가한다고 한다. 즉

$$a_k > \Sigma_{i=1}^{k-1},\ \text{모든 } k=2,\ 3,\ \cdots,\ n.$$

Knapsack 벡터가 초증가하는 성질을 지닐 때 knapsack 문제의 해는 용이하게 구할 수 있다. 즉 n번의 크기 비교와 뺄셈으로 knapsack 문제의 해를 구할 수 있다. 그러나 이는 그 자체로 암호체계가 될 수 없으므로 이를 어려운 knapsack 벡터로 전환시킨다. 무작위하게 초증가하는 knapsack 벡터 b를 선택하고 $\Sigma^n b_i$ 보다 큰 정수 m과 이에 서로소인 정수 w를 임의의 선택한다. 이제 공개키가 되는 knapsack 벡터 a는 b의 각 원소에 모듈라 m으로 w를 곱함으로써 얻어진다.

$$a_k = w\ b_k \bmod m.$$

이제 초증가하는 knapsack 벡터 b에 trapdoor 벡터 a는 공개 암호키가 되며, a로의 전환과정인 m과 w는 비밀키가 된다. 메시지 $x=(x_1, x_2, \cdots, x_n)$는

$$S = ax = \Sigma_{i=1}^{n}\ a_i x_i$$

에 의해 S로 암호화된다.

$$\begin{aligned} S' &= w^{-1} S \bmod m \\ &= w^{-1} \Sigma\ a_i x_i \bmod m \\ &= w^{-1} \Sigma\ (w b_i \bmod m)\ x_i \bmod m \\ &= \Sigma\ b_i x_i \bmod m \\ &= b\ x, \end{aligned}$$

단 $m > \Sigma\ b_i$, $ww^{-1} = 1 \bmod m$.

비밀정보인 w^{-1}과 m을 알고 있는 정당한 수신자는 이들을 이용하여 $S' = w^{-1}S$를 계산하고 초증가하는 벡터 b를 이용하여 $S' = bx$를 풀어서 암호문 S를 복호화할 수 있다.

모듈라 m과 w를 이용하여 knapsack 벡터를 보다 어려운 knapsack 벡터로 전환시키는 과정을 여러번 반복하여 사용할 수 있으며, 일반적으로 그 결과는 한번의 변환과는 동일하지 않다.

다른 공개키 암호체계와 비교하여 knapsack 암호의 가장 큰 장점은 그 암호화 및 복호화의 계산이 매우 용이하다는 것이다. 암호화는 n 미만의 덧셈으로 가능하며, Merkle-Hellman 암호의 경우에는 복호화도 암호화의 경우보다 약 3배 정도의 계산을 더 필요로 할 뿐이다. 예를 들어 블럭의 크기가 n 비트인 RSA 암호에서 암호화 및 복호화 계산의 복잡성은 $0(n^3)$인데 비하여 knapsack 암호는 $0(n^2)$이다. 그러나 knapsack 암호에서는 안전성을 위해서 암호화시 자료의 확장이 불가피하다. 또한 공개키는 어려운 knapsack 벡터로 구성되어 있으므로 공개키의 크기가 매우 커지게 된다. Merkle과 Hellman은 안전상의 관점에서 n이 100 이상이 되어야 한다고 했는데, 이 경우 이들의 제안에 따라 knapsack을 구성할 경우 평문에서 암호문으로의 자료 확장은 약 2.09이고 공개키는 20.2K 비트이다.

Merkle과 Hellman의 knapsack 공개키 암호체계가 개발된 이후 이를 암호분석하는 방법이 많이 개발되었으며 이에 대응하는 새로운 knapsack 공개키 암호체계가 개발되었다. Chinese remainder 정리에 근거한 Lu-Lee 암호체계[6], 대수적 코딩이론에 근거한 Neiderreiter 암호체계[6], 초증가하는

knapsack 벡터가 아니라, Chinese remainder 정리를 이용하여 knapsack 벡터의 radix form과 modular form을 서로 전환시키는데 기초한 쉬운 벡터를 모듈라 곱셈을 이용하여 어려운 knapsack 벡터로 변환시킨 Goodman-McAuley 암호체계[6], GF(2)에서의 다항식에 기초한 Pieprzyk 암호체계 등이 있다. 이들 knapsack 공개키 암호체계중 대부분은 쉬운 knapsack을 구성하고 이를 모듈라 곱셈을 통하여 어려운 knapsack으로 변환시키는 기본적인 개념을 사용하고 있다. 이러한 개념에 근거하지 않는 유일한 knapsack 공개키 암호체계는 Chor와 Rivest의 knapsack 암호체계이다. 이는 유한체에서의 산술에 근거하고 있다.

근래에도 새로운 knapsack 암호체계가 개발되고 있다. 즉 linearly shift 방법을 사용하는 Laih 등의 암호체계[13]와 modular knapsack group 문제에 기초한 Nieme의 암호체계[17]가 그 예이다.

3. Chor-Rivest knapsack 암호체계

Chor-Rivest knapsack 암호체계는 기존의 knapsack 암호체계와 완전히 다른 기초위에 만들어졌다. 이는 초증가 벡터가 아니라 일반적인 knapsack 벡터에 근거하고 있으며, Merkle-Hellman 암호화 같은 정수 연산이 아니라 Cooper-Patterson 암호와 같은 Galois 체 연산을 사용하고 있다. 초증가 knapsack 벡터를 사용하는 것은 그 부분합이 서로 다르기 때문이다. 이는 일반적인 knapsack 벡터에서는 성립되지 않는다. h개의 원소의 합이 서로 다른, 고밀도 knapsack 벡터를 구성할 수 있는가 하는 것에 대해 답이 Bose-Chowla 정리이다. 이 정리에 근거하여 Chor와 Rivest는 knapsack 암호를 구축하였다.

정리 1. (Bose-Chowla 정리)

p가 소수의 승수이고 h>1가 정수이면 다음과 같은 정수 벡터 $\{a_i : 1\le i\le p\}$가 존재한다.

1) $1\le a_i\le p^{h-1}$ $i=1, 2, \cdots, p$.

2) $(x_1, x_2, \cdots, x_p)$와 $(y_1, y_2, \cdots, y_p)$가 서로 다른 비음정수 벡터이고 Σx_i, $\Sigma y_i\le h$이면, $\Sigma a_i x_i \neq \Sigma a_i y_i$이다.

다음의 Chor-Rivest knapsack 공개키 암호체계는 Bose-Chowla 정리에 근거하여 다음과 같이 knapsack 벡터를 구성한다. 이 정리는 바로 다음의 암호체계의 구축과정을 통하여 증명된다.

(1) 소수의 승수인 p와 p 보다 작은 정수 h를 선택한다. 이들은 $GF(p^h)$에서의 이산로그 계산이 용이한 것으로 한다.

(2) GF(p)에서 차수 h인 algebraic $t \in GF(p^h)$를 무작위하게 선택한다. 즉 GF(p)[t]에서 차수가 h인 irreducible monic polynomial f(t)를 선택하고 $GF(p^h)$에서의 연산을 $GF(p)[t]/<f(t)>$로 표현하면 된다.

(3) $GF(p^h)$에서 multiplicative generator g를 임의로 선택한다.

(4) 모든 $i \in GF(p)$에 대하여 $a_i=\log_g(t+i)$를 계산한다.

(5) a_i를 섞는다. : 임의의 순열 $w : \{1, 2, \cdots, p\} \to \{1, 2, \cdots, p\}$를 선택하고 $b_i=a_{w(i)}$로 놓는다.

(6) 임의로 $0\le d\le p^h-2$를 선택하여 $c_i=b_i+d$로 놓는다.

여기서 $\{c_i\}$, p, h가 공개키이며 t, g, w^{-1} d가 비밀키이다. 메시지 $x=(x_1, x_2, \cdots, x_p)$는 원소중에 h개가 1인 0-1 벡터이다. 이는 다음과 같이 암호화된다.

$$E(x)=\Sigma_{i=1}^{p} c_i x_i \bmod p^h-1.$$

이의 복호화 과정은 다음과 같다.

(1) $r(t)=t^h \bmod f(t)$를 계산한다.

(2) 주어진 $S=E(x)$에 대하여 $z=S-hd \bmod p^h-1$를 계산한다.

(3) $q(t)=g^z \bmod f(t)$를 계산한다.

(4) GF(p)[t]에서 차수 h인 다항식 $s(t)=t^h+q(t)-r(t)$를 얻는다.

(5) $s(t)=(t+i_1)(t+i_2)\cdots(t+i_h)$이므로 GF(p)의 원소를 차례로 대입함으로써 h개의 근 i를 발견할 수 있다. w^{-1}로 메시지 x에서 1인 원소의 위치를 알아낸다.

이러한 Chor-Rivest 공개키 암호체계를 구축하는데 가장 큰 문제는 단계 (4)에서 이산 로그 계산이 필요하다는 것이다. 일반적으로 이의 계산은 매우 어렵지만 p^h-1 이 작은 소인수만을 가지는 특수한 경우에는 Pohlig와 Hellman[21]의 알고리즘으로 계산할 수 있다. Chor와 Rivest는 p=197, h=24인 암호체계를 구축하였는데 미니 컴퓨터에서 이산로그의 계산에 약 8시간이 소요되었다고 한다.

Chor-Rivest 암호체계에서 암호화 과정은 p^h 보다 작은 정수 c_i를 h번 더하면 되며 복호화과정은 최대한 2h lop p의 모듈라 곱셈이 필요하다. 공개키의 크기는 p=197, h=24인 경우에 약 40K 비트가 된다. 또한 정보율(Information Rate)은 약 0.556이며 자료의 확장은 1,798이 된다.

f(t)와 w가 비밀이므로, 이 Chor-Rivest 암호체계를 분석적으로 암호분석하는 방법은 brute force 접근방법보다 효율적이지 못하다.

4. Knapsack 암호에 대한 암호분석

Knapsack 암호는 그것이 개발된 이후로 집중적인 암호분석의 대상이 되어 왔으며 많은 경우에 성공적인 결과를 얻었다. knapsack 암호체계의 안전성에 대한 의문이 제기되는 이유중의 하나는 이 암호체계가 근본적으로 선형이라는 점이다. 즉 $\Sigma a_ix_i+\Sigma a_iy_i=\Sigma a_i(x_i+y_i)$이다. 이 선형성에 근거한 암호분석방법이 제시되지는 않고 있지만 일반적으로 선형성은 암호체계의 안전성에 해롭다고 알려져 있다. 또한 Brassard[4]에 의하면 knapsack 암호체계를 암호분석하는 문제가 NP-hard이면 NP=CoNP라고 한다. 만일 NP≠CoNP이라면 일반적인 knapsack 문제의 해를 구하는 것보다 효율적으로 Merkle-Mellman 암호체계를 암호분석할 수 있을 것이다.

한번 변환시킨 Merkle-Hellman 암호체계에 대해 Shamir가 최초로 성공적인 암호분석을 하였다[23]. 그는 Lenstra[15]의 정수 계획법 알고리즘을 이용하여 공개된 knapsack 벡터에서 이를 초증가하는 knapsack 벡터로 변환시키는 (w', m')을 발견하였다. Shamir의 방법은 다른 knapsack 암호에 대한 일반적인 암호분석이 되지 못했는데, Adleman[5]이 Lovasz의 lattice basis reduction 알고리즘이 사용될 수 있음을 발견하고 이를 사용하여 Graham-Shamir knapsack 암호체계를 암호분석하였다. 이 Lovasz[14]의 알고리즘은 knapsack 암호를 분석할 수 있는 중요한 수단이 되었다. 초증가하는 단순 knapsack 벡터를 모듈라 변환시키는 모든 knapsack 암호체계는 이 알고리즘을 사용하여 암호분석되었다.

Lovasz 알고리즘으로 knapsack 암호의 비밀구조를 밝힘으로서 암호분석하는 접근방법이외에 Lagarias와 Odlyzko[12]의 저밀도 knapsack 암호분석방법이 있다. 이는 밀도(density)가 낮은 모든 knapsack 암호에 적용된다. 이와 다른 저밀도 knapsack 암호분석을 Brickell[6]이 제안하였다.

n 차원 실수 공간 R^n에서의 lattice L은 n 개의 basis $\{b_i\}$의 정수선형결합이다. 즉 Z가 정수들의 집합일 때,

$$L=\{\Sigma^n z_ib_i : z_i \in Z\}.$$

Computational Complexity와 정수 이론, 암호학에서의 많은 문제가 lattice에서 0이 아닌 가장 작은 Euclidean norm을 가지는 벡터를 찾는 문제로 전환시킬 수 있다. 이러한 문제는 Simultaneous Diophantine Approximation과 밀접히 연관되어 있다. 이러한 문제가 NP-hard인지는 알려져 있지 않지만, 아직 다항식 시간 알고리즘이 개발되어 있지도 않고 있다. Lovasz의 lattice basis reduction 알고리즘 (L^3 알고리즘)[14]은 lattice의 basis $\{b_i\}$를 가능한한 orthogonal에 가까운 basis $\{b_i\}$로 변환시키는 것이다.

그러면 이 reduced된 basis $\{b_i\}$에서 $b_1 < 2^{n-1} \min \{\|x\|^2 : x \in L, x \neq 0\}$이며 실제로는 모든 reduced basis 벡터가 상당히 작아지는 경향이 있다. 또한 lattice의 basis 내의 원소들이 모두 B 보다 작은 경우에 Lovasz 알고리즘은 $0(n^6(\log B)^3)$의 비트 연산을 필요로 한다. 보다 작은 벡터를 찾으며 계산도 적게 요구되는 Lovasz 알고리즘의 변형이 개발되었다.

Lagarias-Odlyzko 암호분석[12]은 다음과 같은 knapsack 문제의 해를 구하고자 하는 것이다.

$$\Sigma_{i=1}^{n} a_i x_i = S, \quad x_i = 0 \text{ 또는 } 1.$$

이를 위해 우선 다음의 basis로 이루어지는 n+1 차원의 정수 lattice L을 구성한다.

$$b_1 = (1, 0, \cdots, 0, -a_1)$$
$$b_2 = (0, 1, \cdots, 0, -a_2)$$
$$\vdots$$
$$b_n = (0, 0, \cdots, 1, -a_n)$$
$$b_{n+1} = (0, 0, \cdots, 1, S)$$

이 lattice L에서 Lovasz의 basis reduction 알고리즘은 reduced basis를 구하고 이중에 knapsack 문제의 해가 있는지 검사한다. 해를 구하지 못했으면 M을 $M' = \Sigma^n a_i - M$로 대체한 후 위의 과정을 되풀이 한다.

Knapsack 벡터 $a = (a_1, \cdots, a_n)$의 밀도 (density) $d(a)$는 다음과 같이 정의된다.

$$d(a) = n/\log_2(\max_i a_i).$$

Knapsack 암호에서 밀도는 비트들이 전송되는 정보율의 근사치가 된다. Lagarias-Odlyzko의 암호분석방법의 결과는 다음과 같다.

(1) $d(a) < 0.645$인 거의 모든 knapsack 문제의 해는 lattice L에서 가장 작은 비영벡터이다.

(2) $d(a) < (2-\varepsilon)(\log_2 4/3)^{-1} n^{-1} \cong 1/n$인 거의 모든 knapsack 문제의 해는 Lagarias-Odlyzko 알고리즘으로 구할 수 있다.

그러나 실제 검증결과 알고리즘이 좋은 결과를 얻는 최대 밀도 $d_c(n)$는 $1/n$ 이상이 되나, $n \to \infty$일 때 $d_c(n) \to 0$이 된다. Lagarias와 Odlyzko는 $n=50$이고 $\log_2 B = 100$이 되는 knapsack 문제에 대해 Lovasz 알고리즘으로 reduce 하는데 CRAY-1으로 약 14분 소요하였다.

Chor-Rivest knapsack 암호체계에 대해 가능한 암호분석방법이 아직 알려져 있지 않고 있다. 이 암호에서는 밀도가 상당히 높다. 예를 들어 $p=197$, $h=24$인 경우에 밀도는 0.556이었다. 이와같이 knapsack의 밀도가 높은 경우에는 lattice 내에 knapsack 문제의 해가 아니면서 Euclidean norm이 아주 작은 벡터가 많이 존재하게 될 것이므로 저밀도 암호분석은 성공하기 어렵게 된다. $n=102$, $h=12$인 Chor-Rivest 암호에 대한 실제 검증에서도 Lagarias-Odlyzko 암호분석이 거의 유효하지 않았다[7].

5. 정수 계획법에 근거한 암호분석

Lagarias-Odlyzko의 암호분석 방법은 knapsack의 밀도가 낮은 경우에 유효하지만 상대적으로 계산량이 많이 요구된다. 본절에서는 정수계획법의 이완을 이용함으로써 용이하게 knapsack 문제의 부분해를 구할 수 있음을 보인다.

우리가 관심을 가지고 있는 일반적인 knapsack 문제는 다음과 같다.

$$\Sigma_{i=1}^{n} a_i x_i = S, \quad x_i = 0 \text{ 또는 } 1.$$

Lagarias와 Odlyzko에 의하면 이러한 knapsack 문제는 그 밀도가 0.645 보다 작으면 거의 대부분이 다음의 basis를 갖는 lattice에서 Euclidean norm이 가장 작은 비영 벡터를 찾는 문제로 변환될 수 있다.

$$b_1 = (1, 0, \cdots, 0, -a_1)$$
$$b_2 = (0, 1, \cdots, 0, -a_2)$$

.
.
.

$b_n=(0,\ 0,\ \cdots,\ 1,\ -a_n)$

$b_{n+1}=(0,\ 0,\ \cdots,\ 1,\ S)$

즉 다음의 정수 계획법 문제로 변환될 수 있다.

$$\min \|\Sigma_{i=1}^{n+1} z_i b_i\|^2$$
$$\text{s.t. } z_i = \text{정수},$$
$$z=(z_1,\ z_2,\ \cdots,\ z_{n+1})\neq 0.$$

그러나 lattice 벡터가 knapsack 문제의 해가 되기 위해서는 $z_{n+1}=1$이어야 하며, z_i를 0 또는 1로 제약하여도 무관하다. 따라서 위의 문제를 다음과 같이 재변환할 수 있다.

$$\min \Sigma_{i=1}^{n} x_i^2+(S-\Sigma_{i=1}^{n} a_i x_i)^2$$
$$=\Sigma\ (1+a_i^2-2Sa_i)x_i+2\Sigma_{i<j}\ a_i a_j x_i x_j+S^2$$
$$\text{s.t. } x_i=0 \text{ 또는 } 1.$$

여기서 모든 $x_i=0$인 경우는 최적해가 될 수 없으므로 이러한 경우를 배제하는 제약식은 불필요하며, x_i가 0 또는 1이므로 $x_i^2=x_i$이다. 즉 위의 문제는 일반적인 형태의 비제약 0-1 Quadratic Programming 문제이며 이러한 최소화 문제를 최대화 문제로 전환시키면 다음과 같다.

$$\text{(QP)}\qquad \max\ \Sigma_i\ q_{ii}x_i x_i+\Sigma_{i<j}\ q_{ij}x_i x_j$$
$$\text{s.t. } x\in B^n=\{0,\ 1\}^2$$

여기서 $q_{ii}=2Sa_i-1-a_i^2$

$q_{ij}=-2a_i a_j$, 모든 $i<j$.

원래의 knapsack 문제의 해에서 값이 1인 원소가 h개 있을때 문제 (QP)의 최적해는 $h-S^2$ 보다 같거나 크게 된다. 문제 (QP)는 일반적인 maximum-cut 문제로 변환 가능하며 이 경우에 $q_{ij}\geq 0$이면 이 문제는 다항식 시간내에 최적해를 구할 수 있다[19]. 또한 문제 (QP)에서 행렬 $Q=(q_{ij})$가 특수한 구조를 갖고 sparse한 경우에는 쉽게 최적해를 구할 수 있다. 그러나 문제 (QP)는 이러한 유형에 속하지는 않는다.

문제 (QP)는 Rhys[19]에 의하면 다음과 같이 선형화할 수 있다. $z_i=1-x_i$로 정의하면 $q_{ij}x_i x_j$는 $q_{ij}x_i-q_{ij}z_i x_j$로 대체할 수 있으므로 (QP)의 목적식은 다음과 같다.

$$f(x)=-\Sigma_{i<j}\ q_{ij}z_i x_j+\Sigma_i(q_{ii}+\Sigma_{j<i}\ q_{ji})x_j.$$

이제 $z_i x_j$를 새로운 0-1 변수 y_{ij}로 대체하면 (QP)는 다음의 0-1 선형 계획법 문제가 된다.

$$\max\ -\Sigma_{i<j}\ q_{ij}y_i+\Sigma_i(q_{ii}+\Sigma_{j<i}\ q_{ji})x_j.$$
$$\text{s.t. } y_{ij}\leq 1-x_i,\ y_{ij}\leq x_j,\ \text{모든 } i<j.$$
$$x_i\in\{0,\ 1\},$$
$$y_{ij}\in\{0,\ 1\}.$$

위 문제에서 $x_i\in\{0,\ 1\}$, $y_{ij}\in\{0,\ 1\}$의 제약을 없애고 조건 $0\leq x_i,\ y_i\leq 1$을 부가하여 이완한 (relax) 선형 계획법 문제를 Rhys LP라고 하자.

모든 $x\in B^n=\{0,\ 1\}^n$에 대하여 $p(x)\geq f(x)$ 관계가 성립되는 $p(x)$는 $f(x)$의 upper plane이라고 한다. 이는 다음과 같이 (QP)의 선형 이완이 된다.

(1) $\max\ \{p(x)\ :\ x\in B^n\}$.

$f(x)$에 대한 upper plane의 집합인 U는 다음의 관계가 성립할 때 complete한다고 한다.

$$f(x)=\min\ \{p(x)\ :\ p(x)\in U\},\ \text{모든 } x\in B^n.$$

이때 문제 (QP)는 다음 문제와 동일하다.

$\max\{f(x)\ :\ x\in B^n\}$.

문제 (1)의 최적값이, 가능한한 문제 (QP)의 최적값에 가까운 upper plane p(x)를 구하는 것이 중요하다. 즉 이는 다음과 같은 $p^*(x)$이다.

$$(2)\quad \max_{x\in B^n} p^*(x) = \min_{p(x)\in U} \max_{x\in B^n} p(x).$$

이러한 $p^*(x)$는 best upper plane이라고 한다.

다음의 upper plane은 roof라고 하며 이러한 upper plane의 집합을 R이라고 하자.

$$(3)\quad p^*(x) = v_0(\lambda) + v_1(\lambda)x_1 + \cdots + v_n(\lambda)x_n.$$

여기서 $v_0(\lambda) = \Sigma_{i<j}\ \lambda_{ij}$.

$v_i(\lambda) = q_{ii} - \Sigma_{i>j}\ \lambda_{ij} - \Sigma_{j<i}\ \lambda_{ji}$.

위에서 λ_{ij}는 $0 \le \lambda_{ij} \le |\ q_{ij}\ |$ 인 실수이다. 이때 R은 f(x)에 대한 upper plane들의 complete set이 된다. 이제 (3)의 roof를 사용한 (2)의 문제는 다음과 같이 선형 계획법으로 바꿀 수 있다.

$$w(R) = \min_{\lambda\in\Lambda} \max_{x\ B^n} v_0(\lambda) + v_1(\lambda)x_1 + \cdots + v_n(\lambda)x_n$$
$$= \min_{\lambda\in\Lambda} v_0(\lambda) + v_1(\lambda)^+ + \cdots + v_n(\lambda)^+$$

여기서 $a^+ = \max\{a, 0\}$이다. 그러므로

$$(4)\quad w(R) = \min\ v_0 + \Sigma u_i$$
$$\text{s.t. } u_i \ge v_i(\lambda),$$
$$u_i \ge 0, \quad \text{모든 } i=1, \cdots, n,$$
$$0 \le \lambda_{ij} \le |\ q_{ij}\ |. \quad \text{모든 } i<j.$$

실제로 위의 문제는 bidirected flow problem이며 이는 $O(n^6)$ 시간내에 최적해를 구할 수 있다.

$p(x) = v_0 + v_1x_1 + \cdots + v_nx_n$이 f(x)의 best roof이면 문제 (1)의 최적해 x^*는 다음과 같다.

$$x^* = 0 \text{ if } v_i < 0, \ = 1 \text{ if } v_i > 0.$$

정리 2[11].

$p(x) = v_0 + v_1x_1 + \cdots + v_nx_n$이 f(x)의 best roof이고 (x^{**}, y^{**})가 Rhys LP의 최적해이면

$$v_i > 0 \text{ --> } x^{**} = 1,\ v_i < 0 \text{ --> } x^{**} = 0.$$

위의 정리에 의해 모든 best roof에서 v_i의 부호는 비음이던지 비양이던지 동일하게 된다. 또한 best roof에 대한 문제 (1)의 최적 목적식 값과 Rhys LP의 최적 목적식 값은 동일하여 정수 계획법을 연속 계획법으로 이완시킴에 따라 발생되는 gap은 같다.

정리 3[11]. (Strong Persistency Theroem)

만일 f(x)의 어떠한 best roof $\{v_i\}$에서 $v_i>0$ [$v_i<0$] 이면 (QP)의 모든 최적해에서 $x_i=1$ [$x_i=0$]이다.

정리 4[11]. (Weak Persistency Theorem)

만일 Rhys LP의 최적해 (x, y)에서 $x_i=1$ [$x_i=0$] 이면 $x_i=1$ [$x_i=0$]인 (QP)의 최적해가 존재한다.

이제 Rhys 선형화나 roof를 이용하여 원 knapsack 문제의 해를 부분적으로 구할 수 있다. 만일 knapsack의 밀도가 높지 않은 경우에는 거의 모든 경우에 원 knapsack 문제는 정수 계획법 (QP) 문제와 동일하게 되며 Rhys LP나 roof를 사용하는 문제 (4)의 최적해를 구함으로써 정리 2와 같이 Rhys의 최적해에서 $x_i=0$ [$x_i=1$]이거나 $v_i>0$ [$v_i<0$]이 되는 경우에는 그에 해당되는 원래의 knapsack 변수의 값을 알 수 있다.

6. Chor-Rivest 암호체계의 일반화

Chor-Rivest knapsack 암호체계는 기존의 knapsack 암호체계와 달리 모듈라 연산에 근거하지 않고 유한체에서의 이산로그 계산에 근거하며, 초증가

하는 벡터를 이용하지 않고 보다 일반적인 knapsack 벡터를 사용함으로써, 기존의 모든 암호분석 방법에 대하여 안전하다. 그러나 이 암호체계를 구현함에 있어 유한체에서의 이산로그 계산이 필요하므로 효율성이 문제되고 있다. 여기에서는 Chor-Rivest 암호체계가 근거하고 있는 Bose-Chowla 정리를 일반화함으로써 보다 효율적으로 안전한 knapsack 암호체계를 구축하고자 한다.

정리 5. (Bose-Chowla 정리의 일반화)

p가 소수의 승수이고 h>1가 정수이며, g가 GF(p^h) $\cong$GF(p)/<f(t)>의 multiplicative generator라고 하자. 또한 정수 벡터 $\{a_i : 1\le i\le p\}$는 다음의 조건을 만족한다고 하자. (단 $1\le a_i\le p^h-1$)

1) $\{h_i\}$는 서로소이다. (단 $h_i=g^{a_i} \bmod f(t)$)
2) $0<h_i$의 차수 $\le r$, 모든 i
3) $rk<h$

$(x_1, x_2, \cdots, x_p)$와 $(y_1, y_2, \cdots, y_p)$가 서로 다른 비음정수 벡터이고 Σx_i, $\Sigma y_i\le k$이면, $\Sigma a_ix_i\neq\Sigma a_iy_i$이다.

(증명) 만약 $\Sigma a_ix_i=\Sigma a_iy_i$이라고 하면, GF($p^h$)에서

$$g^{\Sigma a_ix_i} = g^{\Sigma a_iy_i}$$

이고, 따라서

$$\Pi g^{a_ix_i} = \Pi g^{a_iy_i}$$

이다. 그러므로 다음의 식이 성립한다.

$$\Pi h_{i(t)}^{x_i} = \Pi h_{i(t)}^{y_i}$$

그러나 조건 (1)과 $x\neq y$에서 이는 모순이다. (QED)

위 정리가 Bose-Chowla 정리와 다른 점은, Bose-Chowla 정리가 단지 GF(p^h)에서 차수가 1인 다항식의 index만을 계산하여 knapsack 벡터로 놓는데 비하여 일반화된 정리에서는 차수가 r 이하인 다항식을 임의로 선택하여 이의 index를 계산함으로서 knapsack 벡터를 구한다는 점이다. 따라서 보다 일반적인 knapsack 벡터를 선택할 수 있다. 그러나 knapsack 벡터의 부분합이 서로 다르게 해주기 위하여, 다항식들은 서로 서로소가 되도록 선택하여 주며 가중치도 h에서 k로 제한 하여야 한다. 따라서 위 정리에서 만일 r=1이고 k=h이면 이 정리는 원래의 Bose-Chowla 정리와 동일하게 된다. 이 일반화된 Bose-Chowla 정리에 근거하여 knapsack 벡터를 구성하는 과정은 다음과 같다.

(1) 소수의 승수인 p와 p 보다 작은 정수 h를 선택한다. 이들은 GF(p^h)에서의 이산로그 계산이 용이한 것으로 한다.

(2) GF(p)에서 차수 h인 algebraic $t\in GF(p^h)$를 무작위하게 선택한다. 즉 GF(p)[t]에서 차수가 h인 irreducible monic polynomial f(t)를 선택하고 GF(p^h)에서의 연산을 GF(p)[t]/<f(t)>로 표현하면 된다.

(3) GF(p^h)에서 multiplicative generator g를 임의로 선택한다.

(4) g^{a_i}의 최대 차수 r과 $rk<h$인 양수 k를 선택한다.

(5) 모든 i에 대하여 $a_i=\log_g(h_i)$를 계산한다.

이제 $\{a_i\}$는 가중치가 k 이하일 때 그 부분합이 서로 다른 knapsack 벡터이다. 각 사용자별로 knapsack 암호체계를 구축하는 과정은 다음과 같다.

(1) p, h, f(t), r, k를 선택한다.

(2) 각 사용자별로 $\{h_i\}$, $\{a_i\}$를 구하고, gcd(m, p^h-1)=1인 m을 선택한다.

(3) 다음의 α_i, ξ를 선택한다.

$\alpha_i=ma_i \bmod p^h-1$,

$\xi=g^{m^1} \bmod f(t)$.

단, $m_1m=1 \bmod p^h-1$.

(4) 임의로 $0\le d\le p^h-2$를 선택하여 $c_i=\alpha_i+d$로

놓는다.

여기서 $\{c_i\}$, p, h, k가 공개키이며 t, ξ, d, $\{h_{ij}\}$가 비밀키이다. 여기서 각 사용자별로 동일한 $\{h_{ij}\}$는 m과 m_1을 써서 감추고 있다. 메시지 $x=(x_1, x_2, \cdots, x_p)$는 원소 중에 k개 이하만 1인 0-1 벡터이다. 이는 다음과 같이 암호화된다.

$$E(x)=\Sigma_{i=1}^{p} c_i x_i \bmod p^h-1.$$

이의 복호화 과정은 다음과 같다.

(1) 주어진 $S=E(x)$에 대하여 $z=S-hd \bmod p^h-1$를 계산한다.

(2) $s(t)=\xi^z \bmod f(t)$를 계산한다.

$$\begin{aligned} s(t)&=\xi^z \bmod f(t)\\ &=g^{m_1 \Sigma ma_{w(i)}} \bmod f(t)\\ &=g^{m_1 m \Sigma a_{w(i)}} \bmod f(t)\\ &=\Pi g^{a_{w(i)}} \bmod f(t)\\ &=\Pi_{i=1}^{k} h_{w(i)} \bmod f(t)\end{aligned}$$

(3) s(t)를 h(t)로 차례로 인수분해함으로써 메시지 x에서 0과 1비트의 위치를 알아낸다.

제안된 암호체계는 한번의 이산로그 계산으로 많은 사용자가 함께 사용할 수 있는 암호체계를 구축할 수 있다는 점에서 Chor-Rivest 공개키 암호체계와 구별된다. 따라서 한번의 이산로그계산만으로 많은 이용자가 함께 사용할 수 있다는 점에서 이 암호체계의 구현이 보다 효율적이다.

제안된 암호체계에서 암호화 과정은 p^h 보다 작은 정수 c_i를 h번 더하면 되며, 복호화 과정은 ξ를 최대한 p^h-1 제곱승하여야 하므로 최대한 2h log p의 모듈라 곱셈이 필요하다. 따라서 전반적으로 4h log

표 제안된 knapsack 암호체계의 구현

i	h_i			$\log_g h_i$
1	3	13	5	12075238
2	5	14	15	9888267
3	14	18	5	13554503
4	9	8	3	4924428
5	27	0	28	8431377
6	24	4	28	20928952
7	10	15	17	15257188
8	0	2	12	2635422
9	28	6	21	14175233
10	1	15	27	25521516
11	19	10	4	25439528
12	7	1	3	16790094
13	15	8	21	25219605
14	22	11	27	7706927
15	20	18	15	21647197
16	19	2	1	24276005
17	25	6	2	10593790
18	17	8	24	25294790
19	1	19	14	4165892
20	3	4	13	22612316
21	20	19	5	17514249
22	30	14	19	3899764
23	16	2	6	10193296
24	25	1	19	25715459
25	22	27	3	16911281
26	30	30	5	11878288
27	23	19	7	14027593
28	30	28	14	9630976
29	29	6	24	4472654
30	25	6	1	20341343
31	11	26	27	21063445

$f(t)=x^5+6x^4+17x^3+18x+11$
$=5x^4+23x^3+20x^2+15x+20$
* : $h(x)$ is written in the form of $h(x)$
$=\alpha_0+\alpha_1 x+\alpha_2 x^2$

p의 GF(p) 연산이 필요하다. 공개키는 k만 추가되므로 공개키의 크기는 Chor-Rivest 암호체계와 거의 동일하게 ph $\log_2$ p가 된다. 또한 비밀 복호키의 크기는 $p^h \log_2 p$이다. 정보율은 다음과 같이 계산된다.

$$R = \frac{\log_2 {}_{p+k-1}C_k}{\log_2 p^h}$$

제안된 암호체계를 구현하는 일예로 p=31, h=5, r=2, k=2인 경우에 정보율은 약 0.361이 되며 이때 밀도 d(c)=1.129이다.

Chor-Rivest 암호체계에서는 f(t)가 알려진 경우에는 임의의 generator g'를 선택하여 이를 밑으로 하여 t+i의 이산로그를 계산한 후, 이것과 $\{c_i - c_1\}$에서 chinese remainder 정리를 사용하여 g와 w를 구할 수 있다. 그러면 나머지 비밀키도 쉽게 구할 수 있게 된다. 그러나 제안된 암호체계에서는 f(t)와 g가 노출되어 있지만 일반적인 다항식 $\{h_i(t)\}$를 사용하므로 이러한 접근방법이 불가능하다. 또한 암호체계의 밀도가 높으므로 Lagarias-Odlyzko 암호분석 방법도 이 암호체계에는 유효하지 않다.

7. 결 론

본 논문에서는 일반적인 knapsack 문제를 0-1 정수 계획법으로 전환시키고 이 문제를 roof 함수를 사용하여 이완시킴으로써 원 knapsack의 부분해를 구할 수 있음을 살펴 보았다. 따라서 이러한 접근방법은 효율적으로 knapsack 암호체계에서 평문의 일부 비트를 알아 낼 수 있게 한다. 그러나 이러한 접근방법이 얼마나 유효할 것인가에 대한 정밀한 고찰이 필요하며 또한 roof 함수를 보다 개선함으로써 그 유용성을 제고하여야 할 것이다.

또한 안전한 것으로 알려진 Chor-Rivest 암호를 변경시킴으로써 보다 효율적으로 암호체계를 구현할 수 있는 방법을 제시하였다. 이는 Bose-Chowla를 일반화하여 한번의 이산로그 계산으로 많은 사용자가 사용할 수 있도록 하는 것이다. 이러한 암호체계가 과연 안전한가에 대한 정밀한 검증이 요구되며, 안전성을 제고할 수 있는 변형방법이 요구된다.

참 고 문 헌

1. 김세헌, 엄봉식, "Knapsack 공개키 암호체계에 대한 암호분석," WISC'90 (1990).

2. Balas, E. and J.B. Mazzola, "Nonlinear 0-1 Programming : I. Linearization Techniques," Mathmatical Programing 30, (1984) pp.1-21.

3. Balas, E. and J.B. Mazzola, "Nonlinear 0-1 Programming : II. Dominance Relations and Algorithms," Mathmatical Programing 30, (1984) pp.22-45.

4. Brassard, C., "A Note on the complexity of Cryptography," IEEE Trans. on Informat. Theory 25 (1979) pp.232-233.

5. Brickell E.F., J.C. Lagarias and A.M. Odlyzko, "Evaluation of the Adleman Attack on Multiply Iterated Knapsack Cryptosystem," CRYPTO'83 (1984) pp.39-42.

6. Brickell E.F., "Solving Low Density Knapsack," CRYPTO'83 (1984) pp.25-37.

7. Chor, B. and R.L. Rivest, "A Knapsack Type Public Key Cryptosystem Based on Arithmetic in Finite Fields," IEEE Trans. on Informat. Theory 34 (1988) pp.901-909.

8. Desmedt, Y.G., J.P. Vandewalle and R.J.M. Govaerts, "A Critical Analysis of the Security of Knapsack Public-Key Algorithms," IEEE Trans. on Informat. Theory 30 (1984) pp.601-611.

9. Dieter, U., "How to Calculate Shortest Vectors in a Lattice," Mathematics of Computation 29 (1975) pp.827-833.

10. Diffie, W., "The First Ten Years of the Public-Key Cryptography," Proceeding of the IEEE

76 (1988) pp. 560-577.

11. Hammer, P.L., P. Hansen and B. Simeone, "Roof Duality, Complementation and Persistency in Quadratic 0-1 Optimization," Mathematical Programing 28 (1984) pp. 121-155.

12. Lagarias, J.C. and Odlyzko, "Solving Low-Density Subset Sum Problems," Journal of the ACM 32 (1985) pp. 229-246.

13. Laith, C.S., J.Y. Lee, L. Harn and Y.K. Su. "Linearly Shift Knapsack Public-Key Cryptosystem," IEEE Journal on Selected Areas in Communications 7 (1989) pp. 534-539.

14. Lenstra, A.K., H.W. Lenstra, Jr. and L. Lovasz, "Factoring Polynomial with Rational Coefficients," Mathematische Annalen 261 (1982) pp. 515-534.

15. Lenstra, H.W., Jr., "Integer Programming with a Fixed Number of Variables," Mathematics of Operations Research 8 (1983) pp. 538-548.

16. Merkle, R.C. and M.E. Hellman, "Hiding Information and signatures in Trapdoor Knapsack," IEEE Trans. on Informat. Theory 24 (1978) pp. 525-530.

17. Niemi, V., "A New Trapdoor in Knapsack," EUROCRYPT'90, (1991) pp. 405-411.

18. Odlyzko, A.M., "Cryptanalytic Attacks on the Multiplicative Knapsack Cryptosystem on Shamir's Fast Signature Scheme," IEEE Trans. on Informat. Theroy 30 (1984) pp. 594-601.

19. Padberg, M., "The Boolean Quadratic Polytope : Some Characteristics, Facets and Relatives," Mathematical Programming 45 (1989) pp. 139-172.

20. Pieprzyk, J.P., "On Public-Key Cryptosystems Built Using Polynomial Rings," EUROCRYPT'85 (1986) pp. 73-78.

21. Pohlig, S.C., M.E. Hellman, "An Improved Algorithm for Computing Algorithm in GF(p) and its Cryptographic Significance," IEEE Trans. on Informat. Theory 24 (1978) pp. 106-110.

22. Schroeppel, R. and A. Shamir, "A $T=O(2^{n/2})$ $S=O(2^{n/4})$ Algorithm for Certain NP-Complete Problems," SIAM Journal on Computing 10 (1981) pp. 456-464.

23. Shamir, A., "A Polynomial Time Algorithm for Breaking the Basic Merkle-Hellman Cryptosystem," IEEE Trans. on Informat. Theory 30 (1984) pp. 699-704.

□ 著者紹介

김 세 헌(正會員)

서울文理大 物理學科 卒業
美 Standford大(經營科學 碩士 및 博士)
美 System control, Inc 社 勤務
現 韓國科學技術院 經營科學科 教授, 本 學會 論文誌 編輯委員長
關心分野 : 컴퓨터 犯罪와 프라이버시 侵害 防止 對策, 情報시스템 保安, 暗號學

엄 봉 식

1985년 서울대학교 경제학과 졸업
1988년 한국과학기술원 경영과학과 졸업(석사)
현 한국과학기술원 경영과학과 박사과정