通信情報保護學會學會誌
第 9 券 第 4 號, 1999. 12

Hidden Subset Sum 문제를 이용한 Chor-Rivest 암호체계

이 희 정*

요 약

Density(밀도)가 비교적 높은 Chor-Rivest 암호체계는 기존의 LLL과 같은 유형의 공격법이 아니라 비밀키를 일부 찾아내므로 써 공격이 가능하고, '98 Crypto에 처음 발표되고 '99 Crypto에 그의 공격법과 안전성이 논의된 hidden subset sum problem은 기존의 knapsack유형의 암호체계와 마찬가지로 밀도가 높을 때 안전하고 밀도가 낮으면 공격이 가능하다. 따라서 두 암호체계의 접목을 통하여 안전한 암호체계가 가능한지를 살펴보는 것도 의미가 있을 것이다. 결론적으로 이야기하면, 두 암호체계의 접목은 여러 가지 문제점을 포함하고 있기 때문에 어려우리라 생각된다. 제1장에서의 hidden subset sum problem을 살펴보고 제2장에서는 Chor-Rivest 암호체계를 분석해보고 제3장에서 Chor-Rivest 암호체계의 변경 가능한 요소들을 살펴보고 제4장에서 Chor-Rivest 암호체계에 hidden subset sum problem의 활용이 가능한지를 살펴보도록 한다. knapsack유형의 암호체계들 중 비교적 최근까지 안전하다고 하는 암호체계들을 살펴봄으로써 이런 유형들의 개발여부를 생각해 볼 수 있는 기회가 되리라 기대된다.

I. Hidden Subset sum Problem

이산대수문제에 기반을 둔 암호체계에서는 x, g^x를 생성해야 한다. 이때 모듈러 곱셈을 해야 하는데 비용이 많이 든다. 그래서 사람들은 모듈러 곱셈 회수를 줄임으로써 비용을 절감하려는 노력을 해왔다. 최초의 시도는 precomputation method를 이용하여 x를 구한 후 g^x를 구하는 것이었다. 그후 Schnorr이 precomputation method를 이용한 pseudo random number generator를 사용하여 x, g^x를 동시에 구하는 방법을 소개하였다.
그러나 이것은 x에대한 임의성의 결여로 정보가 누출되는 약점이 있었다. Rooij는 이점을 이용하여 Schnorr의 방법을 깼고 다시 Schnorr가 보완하여 새로운 방법을 제시했으나 역시 Rooij에 의하여 깨졌다. Crypto '98에서 Boyko, Peinado, Venkatesan이 'hidden subset sum problem'에 기반을 둔 새롭고 간단한 generator 생성법을 소개하였다. 이는 기존의 'subset sum problem'에서 이름을 따온 것이다. 고전적 subset sum problem이란 주어진 집합 $\{a_1, \cdots, a_n \mid a_i \in Z_M\}$과 $b = \sum_{j \in S} a_j$에서 $S \subset \{1, 2, \cdots, n\}$를 찾는 것인 반

* 강남대학교 이공대학 수학과(hjlee@kns.kangnam.ac.kr

** 본연구는 한국정보보호센타의 지원을 일부 받아 수행되었습니다.

면에 hidden subset sum problem이란 $b_i = \sum_{j \in S_i} \alpha_j$ for $i = 1, \cdots, m$ 을 만족하는 $\{\alpha_1, \cdots, \alpha_n\}$를 찾는 것이다. hidden subset sum problem을 이용한 generator 생성 시에는 모듈러 곱셈을 많이 줄일 수 있다. 고전적 subset sum problem의 공격은 주어진 집합 $\{\alpha_1, \cdots, \alpha_n\}$과 b에 의해서 생성되는 lattice에서 가장 짧은 벡터를 찾아내므로 써 가능하다. 이때 밀도(density, $d = \frac{n}{\log_2 M}$)는 낮아야 가능하다. 대표적인 예로 LLL방법을 들 수 있다. 이에 반하여 hidden subset sum problem은 lattice조차 생성할 수가 없다. 따라서 안전할 것이라고 그들은 추측했다. 동시에 이유는 잘 모르겠지만 parameter들이 충분히 커야할 것이라고 생각했다. 그렇다면 스마트 카드와 같은 파라미터가 작아야 하는 곳에는 적절하지가 않고 서버 application에만 가능하고 또 기존의 방법과 비교해서 속도면 에서 얼마나 효율적인가에 대한 의혹을 갖게 되었다. 이에 Crypto '99에서 Nguyen과 Stern은 hidden subset sum problem이 얼마나 어려운 가를 밝혀 냈는데 밀도가 작을 때의 공격법을 소개하고 있고 밀도가 높으면 generator 생성이 uniformly 분포 된다는 것을 보였다. 놀랍게도 hidden subset sum problem에 대한 공격법이 고전적 subset sum problem에 대한 공격법과 전혀 다름에도 불구하고 밀도가 낮아야만 가능하고 밀도가 높으면 가능하지 않다는 사실을 밝혀냈고 따라서 hidden subset sum 기반 문제들은 subset sum 기반 문제들을 일반화한 것임을 알게 되었다.

generator 생성법

Boyko, Peinado, Venkatesan은 몇 가지의 x, g^x생성법을 제안했는 데 그 중에서 가장 간단한 것은 다음과 같다.

n개의 임의의 정수 $\alpha_1, \cdots, \alpha_n$, $\alpha_i \in Z_M$를 생성한다. $\beta_j = g^{\alpha_j}$를 각 j에 대해서 법p로 계산한 후 α_j들과 β_j들을 저장한다. x, g^x의 생성이 필요할 때마다 $S \subseteq \{1, 2, \cdots, n\}$, $|S| = k$를 생성한 후 $b = \sum_{j \in S} \alpha_j \bmod M$을 계산하여 b가 0이면 다시 S를 구하고 그렇지 않으면 $B = \sum_{j \in S} \beta_j$를 법p에 대하여 구한 후 b와 $B = g^b$를 보낸다.

밀도(density)가 작을 때의 공격법

'Hidden subset sum problem'을 다시 말하면 주어진 정수 M과 벡터 $b = (b_1, \cdots, b_m) \in Z^m$에서 $b \equiv \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_n x_n \bmod M$을 만족하는 정수 $\alpha_1, \cdots, \alpha_n$을 찾는 것이다. 이때 벡터 x_i들의 원소들은 모두 0 또는 1이고 $b_1, \cdots, b_m$, $\alpha_1, \cdots, \alpha_n$들은 0에서 M-1까지의 정수이다. 격자 기반 공격은 다음과 같이 할 수 있다. 먼저 $b = \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_n x_n + Mk \bmod M$에서 x_j들과 벡터 k에 의해서 생성되는 lattice, L의 $\overline{L} = (L^{\perp})^{\perp}$을 찾는다. 편의상 x_j들과 k는 일차독립이라고 하자. 다시말 하면 dimension($\overline{L}$)는 n+1이다. dimension이 n+1미만일 때도 $\overline{L}$를 찾을 수 있다. lattice $\overline{L}$을 생성하기 위해서는 조건이 있는데 주어진 b에 대한 직교공간의 줄임 기저(reduced basis)를 찾아서 그 기저의 처음 m-(n+1)개의 벡터가 각 x_j들과 k에 대해서 직교해야만 한다. 이 조건은 결론적으로 $M^{1/n}$이 커야하는데 이것은 밀도($n/\log_2 M$)가 작을 때를 말한다. 다시 말하면, 밀도가 작을 때만이 lattice $\overline{L}$를 찾을 수 있다. 즉, $u \in b^{\perp}$에 대해서

$$u \cdot b = ux_1\alpha_1 + \cdots + ux_n\alpha_n + uMk = 0.$$

따라서 $p_u = (ux_1, \cdots, ux_n, uk)$

$$\perp V_a = (\alpha_1, \cdots, \alpha_n, M)$$

V_a는 m에 대해서 독립이고 따라서 $V_a^{\perp}$도 m에 대해서 독립이다. 만약 u가 작아지면 p_u도 작아진다. $V_a^{\perp}$에 있는 가장 작은 벡터보다 크기를 더 작게 한다면 p_u은 0이 된다. 즉, u와 $\langle x_1, \cdots, x_n, k \rangle$는 직교(orthogonal)한다.

만약 $\{u_1, u_2, \cdots, u_{m-(n+1)}\}$이

$\langle x_1, \cdots, x_n, k \rangle$와 직교하면

$\{u_1, u_2, \cdots, u_{m-(n+1)}\}^{\perp} =$

$\langle x_1, \cdots, x_n, k \rangle$가 된다.

$\{u_1, u_2, \cdots, u_{m-(n+1)}\}$이 $\langle x_1, \cdots, x_n, k \rangle$와 직교하는 조건이 바로 density가 작은 것을 뜻한다.

두번째로, $\overline{L}$로부터 감춰진 벡터 x_j들을 찾는다. 이때 임의로 정한 x_j들의 원소들은 0과1로 구성되어 있으므로 $\overline{L}$에 있는 짧은 벡터들이다. $\overline{L}$의 줄임 기저의 벡터를 살펴보자. 만약 원소들이 {0,1},또는 {0,-1}로 되어 있으면 이것은 $\pm x_j$이다. 그러나 {0,1,-1}로 되어 있다면 x_j보다 짧아진다.

이런 경우를 피하기 위해서 $\overline{L}$를

$L' = 2\overline{L} + Z \times (1, 1, \cdots, 1)$로 변환한다.

이때, 벡터, $2x_j-(1,1,\ldots,1)$은 L'에 속해 있고 원소들은 ± 1로 이루어져 있으며 L'안에는 더 짧은 벡터는 없다. 따라서 어떠한 줄인 기저에도

$\pm\{2x_j-(1,1,\ldots,1)\}$이 나타날 것이다. 이렇게 함으로써 x_j를 찾아낼 수 있다.

이제 b와 x_j를 알고

$b = a_1x_1 + a_2x_2 + \cdots + a_nx_n \mod M$에 서 a_j만을 찾으면 된다. $x_{i,j}$를 x_i의 j번째 원소라 하자. Lattice L은 다음 행렬의 행들에 의해서 생성된다고 하자.

$$\begin{pmatrix} b_1 & x_{1,1} & x_{2,1} & \cdots & x_{n,1} & M & 0 & \cdots & 0 \\ b_2 & x_{1,2} & x_{2,2} & \cdots & x_{n,2} & 0 & M & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \vdots & \vdots & \ddots & \ddots & 0 \\ b_{m'} & x_{1,m'} & x_{2,m'} & \cdots & x_{n,m'} & 0 & \cdots & 0 & M \end{pmatrix}$$

$b - a_1x_1 + a_2x_2 + \cdots + a_nx_n = 0 \mod M$

이기 때문에 $L^{\perp}$은 $(-1, a_1, \cdots, a_n, ?, \cdots, ?)$을 포함한다. $L^{\perp}$의 기저를 찾아서 그것들의 일차결합으로 표현되는 첫번째 원소가 -1인 벡터, $(-1, \beta_1, \ldots, \beta_n, ?, \cdots, ?)$를 찾으면

$(\beta_1, \ldots, \beta_n) \mod M$이 바로 m'개의 연립방정식의 해가 된다. m'이 충분히 크면 유일한 해가 존재하고, $a_1, \cdots, a_n$을 찾을 수 있다. $L^{\perp}$의 기저를 찾는 데에는 다항식 시간이면 충분하다.

II. Chor-Rivest Cryptosystem

비밀키와 공개키 생성과정

$q=p^h$를 정한다. 이때, p는 소수이고 q-1의 약수들은 너무 크지 않게 한다. 그리고 차수가 h인 기약다항식을 정한 후(유한체 이론에서 그런 다항식은 항상 존재한다는 것이 입증되었다.) 이를 f(x)라 하자. t를 f(x)의 근이라 하고 F_q^*는 순환군이기 때문에 생성자가 존재하는데 이를 g라 하자.

$(F[X]/\{f(x)\} \simeq F_q, F_q^* \simeq \langle g \rangle)$.

{0, ⋯,p-1}의 치환 π를 설정한 후에

$g^{b_j} = t + a_{\pi(j)}$가 되는 b_j를 j가 0에서 p-1까지 구한다. 이때 $a_j \in Z_p$에 있고 b_j는 0은 아니고 q-1보다 작다($g^{q-1} = 1$). b_j를 구하는 문제는 이산대수문제인데 다른 암호시스템과는 반대로 이 경우에는 공개키를 만들려고 하는 사람이 이산대수문제를 풀기 쉬워야 한다. 따라서 q-1의 약수들이 작아야지 만 기존의 Pohlig과 Hellman의 알고리즘이나 Coopersmith의 알고리즘을 이용하여 이산대수문제를 풀 수 있다. $v_j = b_j + z$를 구한다. 이때, z는 0에서 p^h-1사이의 정수이다. 이렇게 만든 $v_0, v_1, \cdots, v_{p-1}$와 p,h를 공개하고 비밀키로 z, π, f(x), g,t를 갖고 있다.

암호화 과정

정보를 보내려는 송신자는 메시지를 이진법으로 표현한 후에 메시지 크기를 p비트에 weight h를 갖도록 준비해야 한다. 그 이유는 Bose-Chowler 정리에서 알 수 있다.

Bose-Chowler 정리

p는 소수, $h \geq 2$ 인 정수라 하자. $A = \{a_i \mid 0 \leq i \leq p-1\}$가 다음 2조건을 만족한다고 하자.

1). $1 \leq a_i \leq p^h - 1, \quad i = 0, 1, \cdots, p-1$

2) . 만 약 $(x_0, x_1, \cdots, x_{p-1})$과 $(y_0, y_1, \cdots, y_{p-1})$이 음의 정수가 아닌 원소들로

이루어졌고 서로 다르면서 $\sum_{i=0}^{p-1} x_i, \sum_{i=0}^{p-1} y_i \le h$라고 하자.

그러면 $\sum_{i=0}^{p-1} x_i a_i \neq \sum_{i=0}^{p-1} y_i a_i$.

따라서 p비트에서 weight을 h보다 크게 하면 서로 다른 메시지에서 같은 암호문을 갖게 되어 암호체계로 사용할 수 없다. Bose-Chowler에 의하면 weight이 h보다 작거나 같으면 되는데 Chor-Rivest는 weight을 h로 주고 있다. 메시지를 p비트 h weight로 embedding하는 과정은 널리 알려져 있으므로 생략하도록 한다. 메시지를 $(\varepsilon_0, \varepsilon_1, \cdots, \varepsilon_{p-1})$이라 하자. 이때 ε_i는 0이나 1이고 weight은 h이다. 주어진 공개키 $v_0, v_1, \cdots, v_{p-1}$를 이용하여 $\sum_{i=0}^{p-1} \varepsilon_i v_i = c$를 구하여 암호문을 보낸다.

복호화 과정

수신자는 먼저 알고 있는 비밀키 g와 z를 이용하여 g^{c-zh}를 구한다. g^{c-zh}는 법 f(t)에 대한 t의 다항식으로 표현되어지는데 이를 G(t)라 하자. t대신에 x를 대입하여 $F_p[X]$상에 있는 차수가 h보다 작은 유일한 다항식 G(X)로 나타낼 수 있다. 동시에 $g^{c-zh} = \prod g^{\varepsilon_j b_j} = \prod (t + \alpha_{\pi(j)})^{\varepsilon_j}$이고 따라서 다항식 $\prod (X + \alpha_{\pi(j)})^{\varepsilon_j}$로 나타낼 수 있다. G(X)와 $\prod (X + \alpha_{\pi(j)})^{\varepsilon_j}$, 모두 법 f(x)에 대해서 한 원소를 나타내는 것이나 $\prod (X + \alpha_{\pi(j)})^{\varepsilon_j}$는 차수가 h이고 G(X)는 차수가 h보다 작으므로 법 f(x)에 대해서 이에 해당하는 f(x)를 G(X)에 더하므로 차수를 h로 만들 수 있다. $f(x)+G(X) = \prod (X + \alpha_{\pi(j)})^{\varepsilon_j}$이다. 따라서 f(x)+G(X)를 인수분해 함으로써 $\pi(j)$를 알 수 있고 π^{-1}를 이용하여 ε_j를 찾아 낼 수 있다.

공격법

고전적 subset sum문제와는 달리 LLL과 같은 격자 줄임을 사용할 수가 없다. 따라서 비밀키를 찾아내려고 하는데 우선 일부의 비밀키가 노출되었을 때와 전혀 노출이 되어 있지 않을 때를 생각할 수 있다. 일부의 비밀키가 노출되었을 때는 나머지 비밀키를 찾아낼 수가 있었고 Chor와 Rivest는 비밀키가 전혀 노출이 되어 있지 않을 때 t를 찾으려고 했다. 그러면 그것을 이용하여 다른 비밀키를 찾아낼 수 있기 때문이다. t를 찾는 방법을 제시했으나 실제에 있어서는 거의 실행 불가능하다고 이야기했다. 그후 Vaudenay는 주어진 유한체의 subfield를 이용하여 그곳의 생성자를 찾으므로 써 t를 구하고 비밀키를 구하는 방법을 제시하였다. 따라서 Chor-Rivest 암호체계는 기존의 Kanpsack 암호체계보다는 안전하지만 역시 비밀키에 대한 공격으로 인하여 안전하지 못하다고 인식되고 있다.

'hidden subset sum problem'에 기반을 둔 새로운 공개키 암호알고리즘을 만든다면 density가 높을 때 현재까지는 공격에 안전하기 때문에 유용할 것이다. 또한 밀도가 높은 Chor- Rivest Knapsack 알고리즘에 'hidden subset sum problem'을 변형시킬 수 있다면 강력한 알고리즘이 되리라 생각된다. 이를 위해서는 먼저 Chor-Rivest 암호체계의 변형 가능한 요소들을 살펴보고 나타나는 문제점들을 생각해 본다. 그후 Chor-Rivest 암호체계에 어떻게 'hidden subset sum problem'을 활용할 수 있는가를 살펴보도록 한다.

III. Chor-Rivest 암호체계의 변경 가능한 요소들 분석

Chor-Rivest 암호체계에서 제일 먼저 생각할 수 있는 것은 weight h이다. Bose-Chowler에 의하면 weight을 h보다 작거나 같으면 된다고 했는데 Chor-Rivest는 weight를 h로 했다. 이곳을 변경할 수 있는 여부를 살펴볼 필요가 있다. 사용자에게 메시지의 weight를 h이하 임의로 하도록 해 보자. 우선 다른 메시지에 대해서 암호문이 같아질 우려는 할 필요가 없다. 그러나, weight이 h보다 작으면 전달하려는 정보의 양도 줄어든다. 따라서 사용자 입장에서는 유리하지가 않다. 더구나 수신자도 weight을 모르기 때문에 복호화 과정에서 얼마만큼의 비밀키 z를 빼야 하는지 알 수가 없다. 이런 문제를 해결하기 위해서 공개키를 작성할 때 상수 z를 더하지 말고 permutation만 한 상태에서 공개하면 가능할 것이다.($v_i = b_i$) 그러나, 상수만큼의 변환을 하는 것과 하지 않는 것과의 차이는 안전도에 있어서는 큰 차

이가 없다. 비밀키 하나가 줄어서 암호화와 복호화 과정을 간편히 한 측면과 사용자에게 임의성을 더 많이 부여한 장점은 있으나 안전성에 있어서 불안(심리적)만 가중시켰고 정보량에 있어서도 감소되었기 때문에 weight을 임의로 정하는($\leq h$) 변형은 크게 관심을 받지 못할 것이다.

다음으로 생각해 볼 수 있는 것은 $g^{b_i}=t+\alpha_{\pi(i)}$ 에서 $t+\alpha_{\pi(i)}$를 2차 이상으로 바꿀 수 있는가 하는 문제이다. 만약 가능하다면 b_i의 개수가 늘어나기 때문에 density($\frac{n}{\log_2 M}$)을 크게 할 수 있고 따라서 안전성을 높일 수 있을 것이다. 가장 간단한 $t^2+a_it+b_i$ $a_i, b_i \in Z_p$를 생각해보자. p^2개의 b_i를 찾을 수 있다. 그러면 여기서 몇 가지 근본적인 문제들을 [illegible] 우선, p^2개의 b_i들 가운데에서 h개를 [illegible] 과연 다 다른 값을 가질 것인가 하는 문제이다. 이것이 가능하다면 기존의 방법에 비해서 안전도를 높일 수 있다. 아니면 weight의 변화를 주어 암호체제를 유지할 수 있을까, weight의 증가는 가능한가를 생각해 볼 수 있다. 결국 다음과 같이 Bose-Chowler정리를 확장할 수 있는가 하는 문제이다.

Bose-Chowler 정리의 확장

p는 소수, $h \geq 2$ 인 정수라 하자. $A=\{a_i \mid 0 \leq i \leq \quad 1\}$가 다음 2조건을 만족한다고 하자.

1). $1 \leq a_i \leq p^h - 1, \quad i=0,1,\cdots,p^2-1$

2) . 만 약 $(x_0, x_1, \cdots, x_{p^2-1})$과 $(y_0, y_1$ [illegible] 음의 정수 [illegible] 원소들로 이루어졌고 또 다르면서

$\sum_{i=0}^{p^2-1} x_i, \sum_{i=0}^{p^2-1} y_i \leq [h/2]$라고 하자. 그러면,

$\sum_{i=0}^{p^2-1} x_i a_i \neq \sum_{i=0}^{p^2-1} y_i a_i$.

증명:

$\sum_{i=0}^{p^2-1} x_i a_i = \sum_{i=0}^{p^2-1} y_i a_i$이라고 가정하자.

즉, $g^{\sum_{i=0}^{p^2-1} x_i a_i} = g^{\sum_{i=0}^{p^2-1} y_i a_i} \mod p^h$

$$\prod_{i=0}^{p^2-1} (g^{a_i})^{x_i} = \prod_{i=0}^{p^2-1} (g^{a_i})^{y_i}$$

$$\prod_{i=0}^{p^2-1} (g^{a_i})^{x_i} = \prod_{i=0}^{p^2-1} (t^2+b_{i1}t+b_{i2})^{x_i}$$

$$= \prod_{i=0}^{p^2-1} (g^{a_i})^{y_i} = \prod_{i=0}^{p^2-1} (t^2+c_{i1}t+c_{i2})^{y_i}.$$

$b_{i1}, b_{i2}, c_{i1}, c_{i2} \in Z_p$

0이 아닌 x_i, y_i에 대해서 weight 계산을 하면 $\sum_{i=0}^{p^2-1} 2x_i, \sum_{i=0}^{p^2-1} 2y_i \leq 2[h/2] = h$ 가 된다. 이때, t에 대해서 정리하여 한 방향으로 옮기면 양쪽이 모두 monic 다항식이기 때문에 차수가 하나 줄어든다. 이것은 t가 차수가 h인 기약다항식의 근이라는 사실에 위배된다. 따라서, 서로 다른 두 벡터 $(x_0, x_1, \cdots, x_{p^2-1})$, $(y_0, y_1, \cdots, y_{p^2-1})$은 weight으로 h/2보다 크지 않은 정수를 주면 서로 다른 값, $\sum_{i=0}^{p^2-1} x_i a_i$, $\sum_{i=0}^{p^2-1} y_i a_i$을 갖게 된다. ■

위의 정리를 통하여 2차항 이외에 3차항으로 확장할 수 있고 여러 차수의 형태로 확장할 수 있을 것이다. 단지 차수의 합이 h를 넘지 않도록 weight을 주면 될 것이다. 2차 이상에는 weight이 줄기 때문에 density는 커지지만 전달하려는 정보의 단위는 줄어들 것이다.

또다른 문제는 이렇게 확장했을 경우에 복호화는 어떻게 할 것인 가이다. 복잡함을 피하기 위해서 permutation은 생략하기로 하자. 주어진 $A=\{a_i \mid 0 \leq i \leq p^2-1\}$로부터 암호화된 메시지 $E(m)=\sum_{i=0}^{p^2-1} x_i a_i$를 받았을 때

$$g^{E(m)} = \prod_{i=0}^{p^2-1} (g^{a_i})^{x_i} = \prod_{i=0}^{p^2-1} (t^2+b_{i1}t+b_{i2})^{x_i}$$

동시에 $g^{E(m)}$은 t에 관한 h-1이하의 다항식으로 표현된다. 따라서 법 f(t)에 대해서 0에 해당하는 f(t)를 $g^{E(m)}$에 더하여 t대신에 x를 대입한 후 인수분해를 한다. 이 경우는 1차항의 경우와는 다르게 복잡해진다. 2차 기약다항식인 경우도 있지만 인수 분해되어 1차항으로 인수분해 되는 경우도 있으므로 메시지 위치, (x_i)를 찾아내는 데에 어려움이 예상된다. 메시지 위치가 유일하게 나타나지 않고 중복되는지, 만약 그러한 결과가 초래될 때 어떻게 복호화 가능하게 변형할 수 있는지, 전혀

불가능한 지에 대한 연구는 다음으로 미루도록 한다.

다음에 생각할 수 있는 것은 이산대수로 찾아놓은 $\log_g(t+\alpha_j)$들의 변환을 permutation과 상수만큼의 변환 외에 다른 방법은 없겠는가 하는 것이다. 가장 자연스러운 것은 knapsack일 것이다. 그러나 이에 대한 논의는 hidden subset sum에서 다루기로 하자. 그 외에는 다시 새로운 생성자 g'을 찾아서 이산대수문제를 활용하는 것이다. 즉,

$g'^{b_i} = c_i \mod p^h$를 하여 c_i를 공개하는 것이다. 그러나 이 방법은 공격에 취약하다. 그 이유는 b_i를 구하기 위하여 q-1의 약수를 작게 했기 때문에 c_i에서 b_i를 구할 때에도 쉬울 것이기 때문이다.

IV. Hidden Subset Sum 문제 활용 분석

Hidden subset sum을 Chor-Rivest암호체계에 활용하기 위해 제일 먼저 생각할 수 있는 것은 이산대수 문제, $g^{b_j} = t + \alpha_{\pi(j)}$에 이용하는 것이다. $t+\alpha_{\pi(j)}$에 해당하는 b_j를 찾는 것이 아니고 Boyko, Peinado, Venkatesan가 제안한 것처럼 hidden subset sum 문제를 이용하여 g^{b_j}를 임의적으로 구한다면 g^{b_j}는 더 이상 일차인수의 곱이 되지 않으므로 메시지를 복호화 하기가 어렵다(기존의 방법으로는). 더구나 비밀키 b_j를 공개하는 역체계이므로 적합하지가 않다.

다음으로 생각할 수 있는 것은 Chor-Rivest 암호체계와 같이 역 이산대수문제를 이용하여 b_j를 구한 후 p개의 b_j들을 hidden subset sum문제를 이용하여 새로운 c_j를 만들어 공개하는 것이다. 그러나 여기서는 임의의 b_j에서 출발하는 것이 아니고 $t+\alpha_{\pi(j)}$에서 온 b_j라는 정보가 노출되어 있다. 또 다른 문제는 weight이다. 정당한 송신자가 weight h로 메시지를 보내면 수신자는 $E(m) = \sum_{j=0}^{p-1} \varepsilon_j c_j$로부터 $E(m) = \sum_{j=0}^{p-1} \varepsilon_j c_j = \sum_{j=0}^{p-1} \varepsilon_j \sum_{i \in S_i} b_{ji}$를 얻는다. 이때 b_j들이 중복되므로 더 이상 weight이 h가 되지 않는다. 이를 피하기 위해서는 p개의 b_j들이 분할되어야만 이 중복을 피하고 따라서 일차식의 중복을 피할 수 있어서 메시지를 복호화 할 수 있다. 그러나 분할을 할 경우에는 더 이상 hidden subset sum문제가 아니고 Webb의 complementing subsets 문제가 될 것이다. 설사 complementing subsets을 이용한 변환을 하더라도 lattice reduction을 이용하여 주어진 $v_j's$들에서 $b_j's$를 찾는 것은 쉬울 것이고 그 다음에는 결국 비밀키를 일부 찾아야 하는 기존의 Chor-Rivest 암호체계의 안전성과 같아진다.

V. 결 론

Hidden subset sum 문제는 density가 높을 때 공격법이 아직까지는 알려지지 않았다. 따라서 이 문제에 기반을 둔 새로운 암호체계를 개발할 수 있다면 안전한 암호체계가 될 수 있을 것이다. 그러나, Chor-Rivest 암호체계에 활용하는 문제는 본 연구 결과로는 어려우리라 예상된다. 또한, Chor-Rivest 암호체계의 변형 가능성 여부도 살펴보았으나 이것도 부정적이다. 결론적으로 knapsack기반 암호체계(고전적 의미의 subset sum 문제나 hidden subset sum 문제)는 density에 의해서 안전성이 결정되기 때문에 높은 density에서만 안전한 암호체계를 생산할 수 있다고 말할 수 있다.

참 고 문 헌

[1] V.Boyko,M.Peinado and R.Venkatesan, *Speeding up discrete log and factoring based schemes via precomputations*, Proc. of Eurocrypto'98, Vol.1403 of LNCS, pp.221 -235, Springer-Verlag, 1998.

[2] Phong Nguyen, Jacques Stern, The Hardness of the Hidden Subset Sum Problem and its Crytographic Implications, Proc. of Crypto'99, Springer-Verlag, 1999 ,Aug.

[3] B.Chor, R.L.Rivest, A Knapsack-type

Public Key Cryptosystem based on Arithmetic in Finite Fields, In Advances in Cryptology Crypto'84, Santa Barbara,Cal., Lectures Notes in Computer Science, pp.54 -65, Springer-Verlag, 1985.

[4] Serge Vaudenay, Cryptanaysis of the Chor-Rivest Cryptosystem, pp. 243 -256, 1997(8).

[5] Menezes, Oorschot, Vanston, Handbook of Applied Cyptography, CRC Press, 1997.

[6] N.Koblitz, A Course in Number Theory and Cryptography, 2nd.ed., Springer, 1994.

[7] N.Koblitz, Algebraic Aspects of Cryptography, Springer, 1998

著者紹介

이희정(Hee Jung Lee) 정회원

1980년 2월 : 이화여자대학교 문리대 수학과 졸업

1989년 8월 : 펜실베니아 주립대학교 수학과 박사

1994년3월~현재 : 강남대학교 수학과 조교수

〈관심분야〉 암호학, 유한체이론, 정수론