동형 McEliece 암호

조진규*, 김영식**, 노종선*
*서울대학교 전기정보공학부 뉴미디어통신공동연구소
**조선대학교 정보통신공학과
*jgjo114@ccl.snu.ac.kr, **iamyskim@Chosun.ac.kr, *jsno@snu.ac.kr

Homomorphic McEliece Encryption

Jinkyu Cho*, Young-Sik Kim**, Jong-Seon No*
*INMC, Department of ECE, Seoul National Univ.
**Department of ICE, Chosun Univ.

요 약

본 논문에서는 암호문에 대한 동형 덧셈 연산이 가능한 McEliece 암호 시스템을 새롭게 제안하였다. 이진 기약 Goppa 부호를 사용한 McEliece 암호 시스템은 양자 컴퓨터 상의 연산에서도 현재까지 해독되지 않는 포스트 양자 암호의 대표적인 후보 중 하나이다. 공개키의 크기가 기존 RSA 암호나 타원곡선 암호에 비해 너무 크다는 단점을 갖고 있지만, 암호화나 복호화의 복잡도가 낮아 활용 가능 분야는 더 넓을 것으로 기대되고 있다. 특히 IoT와 같은 제한된 연산량을 갖는 장치에서도 적용 가능한 것으로 알려져 있다. McEliece 암호에서는 랜덤 오류가 더해지기 때문에 암호문 생성 시 부호화 과정 자체는 선형으로 수행되지만 오류가 더해진 후에는 선형의 특성을 잃어버리게 된다. 따라서 기존의 방식에서는 McEliece 암호를 사용해서는 동형 연산이 덧셈에 대해서도 불가능하였다. 본 논문에서는 McEliece 암호가 덧셈에 대해 동형이 되기 위해서 암호화 단계에서 오류 생성을 위한 $\phi_{n,t}$이 가져야 할 조건을 제시하며 이러한 조건을 만족할 수 있는 $\phi_{n,t}$을 제안한다.

Ⅰ. 서 론

지금까지는 RSA 또는 타원곡선기반 공개키 암호시스템이 기반하고 있는 큰 정수의 인수분해 문제나 이산 로그 문제가 전통적인 컴퓨터 상에서 다항 시간 안에 해독할 수 없었기 때문에 이들이 널리 사용되었다. 하지만 Shor의 알고리즘에 의해 양자 컴퓨터를 사용하면 다항 시간 안에 해독할 수 있을 뿐만 아니라 최근 양자 컴퓨터의 상용화가 점점 가시화되면서 RSA나 타원곡선암호를 대체할 수 있고 양자 컴퓨터를 이용한 공격에도 위협을 받지 않는 포스트 양자 암호 시스템들에 대한 연구가 집중되고 있다. 그러한 암호로는 부호 기반 암호, 격자 기반 암호, 해쉬 기반 암호, 다변수 기반 암호 등이 있다. 그 중, 부호 기반 암호는 오류정정부호를 활용한 암호로 랜덤한 선형 부호의 신드롬 디코딩이 NP-Complete이라는 사실에 기반을 두고 있는 암호 시스템이다.

최초의 부호 기반 암호는 1987년에 McEliece가 제안한 공개키 암호 시스템으로,[3] 그 뒤를 이어 다른 종류의 오류 정정 부호를 이용한 암호 시스템들이 계속 제안되었다. 또한 Niederreiter는 패리티 검사 행렬을 기반으로 둔 다른 형태의 부호기반 암호를 제안하였으며, 이 방식이 McEliece 암호와 등가라는 사실이 알려져 있다. 공개키 암호화뿐만 아니라 부호 기반 전자 서명 방식으로 CFS 전자서명도 제안되었으며, Stern의 식별 도식(identification scheme) 등이 개발되기도 하였다.[1]

부호 기반 암호에서는 주로 Goppa 부호, GRS 부호, LDPC 부호, Reed-Muller 부호 및 Rank metric 기반 Gabbidulin 부호 및 LRPC 부호를 사용한다. 본 논문에서는 덧셈에 대해 동형 연산이 가능한 새로운 형태의 McEliece 암호 시스템을 새롭게 제안한다. 이를 이용하여 클라우드 상에서 암호문을 해독 없이 직접 덧셈 연산이 가능하게 되어 클라우드 상의 프라이버시 보호 알고리즘으로 사용하는 것이 가능하다.

Ⅱ. 본론

McEliece 공개키 암호는 일반적으로 최소 거리가 $d \geq 2t+1$인 임의의 이진 기약 Goppa 부호의 복호 알고리즘 및 생성행렬 랜덤화에 사용된 Scrambling 행렬과 순열 행렬을 개인키로 사용하고, Scrambling 행렬과 순열 행렬을 사용해서 랜덤화시킨 생성 행렬을 공개키로 사용한다. 암호화 과정에서 랜덤화된 생성행렬을 통해 만든 부호어(codeword)에 해밍 무게 t 이하의 랜덤 오류 값을 더하여 암호문을 만들게 된다.

McEliece 암호는 공개키의 크기가 기존의 RSA 및 타원 곡선 암호에 비해 매우 크다는 단점을 갖고 있다. 그러나 IND-CCA2 보안을 위해 제안된 Kobara-Imai의 메시지 변환 방법과, Systematic 생성 행렬을 사용하게 되면, 공개키의 크기를 기존의 nk에서 $k(n-k)$로 줄이는 것이 가능하다. 이 과정은 다음 <표 1>에 기술되어 있다.

- **공개키**: $k \times (n-k)$ 이진 행렬 R
- **개인키**: $(I_d|R)$에 의해 생성된 부호 g의 복호 알고리즘 D_g
- **암호화**: $y = (z_1, z_1 R) + \phi_{n,t}(z_2)$
  - $y \in \{0,1\}^n$: 암호문
  - $m \in \{0,1\}^{k+l}$: 평문(메세지)
  - r: 랜덤
  - $u_1 = Hash(r) \oplus (m \| padding)$

$u_2 = (r \oplus Hash(u_1))$

$z_2 \| z_1 = u_2 \| u_1,\ |z_1| = k,\ |z_2| = |u_2| + |u_1| - k$

$\phi_{n,t}(z_2) \in \{0,1\}^n$: 오류

- **복호화:** $\underline{x = (x_1, x_1 R) = D_g(y)}$

$(z_2, z_1) = (x_1, \phi_{n,t}^{-1}(y - x))$, $u_2 \| u_1 = z_2 \| z_1$

$r = u_2 \oplus Hash(u_1)$,

$m \| padding = u_1 \oplus Hash(r)$

<표 1 혼합 McEiliece 암호 알고리즘의 γ-conversion>

기존의 McEliece 시스템과 다른 점은 두 가지가 있다. 하나는 발생 행렬을 기존의 G 대신, Systematic 생성 행렬 $G_{syst} = (I_d | R) = UG$ (U는 정칙 행렬)를 사용한다는 점이고, 다른 하나는 Niederreiter 공개키 암호 시스템에서와 같은 방법으로 $\phi_{n,t}$를 이용해서 메시지로부터(여기서는 m_2로부터) 오류를 1 비트만큼 생성해 낸다는 것이다.

$\phi_{n,t}$를 사용하면, 메시지로 오류를 표현해내기 때문에 공개키의 크기가 줄어든다는 장점이 있지만 오류의 분포를 파악할 수 있기 때문에 의미론적인 안전성(semantic security)이 떨어질 수 있다. 하지만 이때, 암호화 과정을 $\psi_G(m,e) = mG + e$라고 하면, 아래의 식과 같이 모든 ψ_G를 $\psi_{G_{syst}}$을 이용해 나타낼 수 있고, 따라서 m_2가 균일하게 분포되어(uniform distribution) 있다면, 기존의 McEliece 암호 시스템과 동일한 안전성을 가진다.[1]

$$\psi_{G_{syst}}(mU^{-1}, e) = (mU^{-1})(UG) + e = mG + e = \psi_G(m,e)$$

추가적으로 이 시스템에서 $\phi_{n,t}$를 동형 함수로 사용하면 훨씬 더 효과적인 암호 시스템이 될 것으로 보인다. 완전 동형 암호에서는 평문 $\Pi_1, ..., \Pi_t$를 암호화할 때, 개인키를 가진 사람을 포함한 모든 사람이 $f(\Pi_1, ..., \Pi_t)$를 암호화한 암호문에 접근할 수 있다. 하지만 $\Pi_1, ..., \Pi_t$는 물론이고, $f(\Pi_1, ..., \Pi_t)$를 포함한 평문에 관한 어떠한 중간 정보도 유출되지 않는다.(단, 여기서 f는 계산 가능한 모든 함수다.)[2]. 이 때, 덧셈에 대한 동형 암호는 f가 비트에 대한 덧셈인 경우이다.

$$\begin{array}{l} \Pi_1 \xrightarrow{\text{암호화}} C_1 \\ \cdot \quad \cdot \quad \cdot \\ \cdot \quad \cdot \quad \cdot \\ \cdot \quad \cdot \quad \cdot \\ \Pi_t \xrightarrow{\text{암호화}} C_t \end{array} \qquad f(C_1, ..., C_t) \xrightarrow{\text{복호화}} f(\Pi_1, ..., \Pi_t)$$

따라서 동형 암호를 사용하면 평문에 대한 정보는 유출되지 않으면서 평문을 암호문으로 만들고, 그 암호문들을 원하는 대로 변형하고 복구시킬 수 있다. 따라서 시스템 개발자가 자신이 원하는 대로 조절할 수 있는 암호 시스템이 되는 것이다. 하지만 부호 기반 암호에서 $\phi_{n,t}$를 동형 암호로 만드는 것은 쉽지 않다. $c_1 = m_1 + e_1, c_2 = m_2 + e_2$에서 $c_1 + c_2$를 해주면 오류가 $e_1 + e_2$만큼이 되기 때문에 오류의 크기는 $2t$보다 작거나 같아진다. 즉, 오류의 무게가 t 이상이 되므로 오류 정정이 불가능해진다.

하지만 이 때, $\phi_{n,t}$를 오류의 무게를 t 이하로 조절할 수 있는 특수한 형태의 선형 부호를 사용하면 동형 McEliece 암호를 만들 수 있다.

일정 무게 부호(constant weight code)를 사용하면 모든 부호어의 해밍 무게가 일정하기 때문에 부호어들의 합은 일정한 무게를 가지게 된다[4]. 따라서 이를 이용하면 동형 암호가 되면서 오류의 무게를 t이하로 만들 수 있다. 아래에 (7,4) 해밍 부호를 예로 들어 보면 여기에서 어떤 두 부호를 임의로 더해도, 또, 세 부호를 모두 더해도 그 무게가 항상 4가 된다. 따라서 $\phi_{n,t}$를 통해 무게가 4 이하인 오류를 만들 수 있고 오류 정정 능력이 4 이하라면, 복호가 가능하게 되는 것이다. 더 큰 무게의 오류를 정정하기 위해서는 더 큰 크기의 일정 무게 부호가 필요하다.

$$c_1 = (0111|100)$$
$$c_2 = (1011|010)$$
$$c_3 = (1101|001)$$

이와 비슷한 방법으로 무게가 두 가지 종류 밖에 없는 코드를 사용하는 방법도 있다.[5] 이때 이 두 가지 중에 큰 무게가 t보다 작도록 만들면 일정 무게 부호와 마찬가지로 구현이 될 것으로 보인다.

Ⅲ. 결론

본 논문에서는 McEliece 암호 시스템의 특징과 단점, 그리고 개선 방안에 대해서 논의해보았다. McEliece 암호 시스템은 굉장히 오래 전에 제안된 방법이지만 아직까지 양자 컴퓨터를 포함한 어떠한 위협적인 공격도 발견되지 않았다. 공개키의 크기가 크다는 단점이 있지만 본 논문에서 논의한 것과 같이 체계적인 발생 행렬을 사용하고 $\phi_{n,t}$를 이용하면 어느 정도 보완이 가능하다. 또한 동형 암호를 적용하게 된다면 더 효과적인 암호 시스템이 될 것으로 보인다. 아직은 완성되지는 않았지만 IND-CCA2 안전하면서 $\phi_{n,t}$가 동형 암호인 시스템이 만들어진다면 양자 컴퓨터가 상용화된 이후에도 꾸준히 사용할 수 있는 효과적인 암호 시스템으로 발전할 것으로 기대된다.

ACKNOWLEDGMENT

이 논문은 2016년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임(R-20160229-002941, IoT 및 클라우드 컴퓨팅을 위한 경량 포스트 양자 암호 시스템 연구)

참 고 문 헌

[1] Bhaskar Biswas and Nicolas Sendrier, "McEliece Cryptosystem Implementation:Theory and Practice", pp. 48-52, Jan. 2009.

[2] Gentry, Craig, A Fully Homomorphic Encryption Scheme, UMI, pp. 27-42, Sep. 2009.

[3] R. J. McEliece "A public key cryptosystem based on algebraic coding theory," DSN progress report, pp. 114-116, 1978.

[4] A.E.Brouwer, James B.Shearer, N.J.A.Sloane and Warren D.Smith, "A New Table of Constant Weight Codes.", pp. 1334-1338, Nov.1990

[5] Ziling Heng and QinYue, "A construction of q-ary linear codes with two weights.", Journal of Latex Templates, pp.1-14, June 2017

[6] Kazukuni Kobara and Hideki Imai, "Semantically Secure McEliece Public-Key Cryptosystems-Conversions for McEliece PKC", PKC 2001, pp.24-28, 2001