한국통신학회논문지 '13-05 Vol.38C No.05
http://dx.doi.org/10.7840/kics.2013.38C.5.428

키 전환이 필요 없는 완전 준동형 암호화 기법

김 재 헌◆°, 유 상 경*, 이 상 한**

Fully Homomorphic Encryption Scheme without Key Switching

Jae-Heon Kim◆°, Sang-Kyung Yoo*, Sang-Han Lee**

요 약

본고에서는 키 전환(key switching) 과정이 필요 없는 Ring-LWE(Learning With Errors) 기반 완전 준동형 암호화(FHE : Fully Homomorphic Encryption) 스킴을 제안한다. 기존의 LWE 기반 FHE 스킴은 벡터 공간의 원소인 암호문의 차원을 줄이기 위하여 키 전환(key switching) 과정을 필요로 하였다. 이 key switching 과정은 새로운 개인키/공개키 쌍과 부가적인 연산 과정을 필요로 하여 FHE 스킴 구현에 있어서 구현 효율성 저하의 큰 요인이 된다. 우리는 환(ring) 상의 이차방정식을 푸는 문제의 어려움이라는 새로운 안전성 가정을 이용하여 암호문의 차원을 줄임으로써 키 전환 과정이 필요 없는 FHE 스킴을 제안한다. 이 방법은 기존의 키 전환 과정에 비해 필요로 하는 새로운 공개키 크기가 매우 작고 부가 연산이 거의 없다는 측면에서 FHE 구현 효율성을 제고할 수 있다.

Key Words : FHE, Key Switching, LWE, Cloud, Security

ABSTRACT

We present a fully homomorphic encryption (FHE) scheme without key switching based on ring- learning with errors (RLWE) problems and some other assumption. Previous FHE schemes based on LWE needed a step called key switching to reduce the dimension of ciphertext. The key switching step actually needs a heavy computation and severe increasement of keys. So the key switching step is a big burden for implementing FHE Schemes. We suggest a FHE scheme without key switching step by reducing the dimension of ciphertexts in other way. Instead of throwing away key switching, we need another hardness assumption of the difficulty of solving quadratic equation over rings.

Ⅰ. 서 론

IoT(Internet of Things) 환경에서는 다양한 사물에 통신기술이 접목되어 사물과 사물 사이의 네트워크를 구성하고 수집/전달받은 정보를 처리하여 고차원의 정보로 가공할 수 있다. 클라우드 컴퓨팅 환경은 현재 개인 컴퓨터가 제공하는 모든 작업 환경을 클라우드 서버를 통해 서비스로 제공하고자 한다. 따라서 IoT 환경에서의 정보의 저장과 처리를 네트워크에 연결된 클라우드 서버에 위탁하는 방법으로 운용 성능을 높이고 IoT가 적용될 수 있는 범위를 넓힐 수 있다[1].

하지만 이러한 환경이 구축되면 필연적으로 사용자 프라이버시 노출 위협이 발생하게 된다. 이를 방어하기 위한 한 가지 좋은 방법은 클라우드에 데이터를 모두 암호화한 후에 암호화한 상태로 모든 연산을 하는 것이다. 이러한 배경에서 데이터가 암호화된 상태로 연산을 가능하게 하는 준동형 암호(HE : Homomorphic Encryption)는 클라우드 컴퓨팅 환

◆° 주저자 겸 교신저자 : ETRI 부설연구소, jaeheon70@gmail.com, 정회원
* ETRI 부설연구소, 한국과학기술원 전기 및 전자공학과, skyoo94@kaist.ac.kr, 정회원
** ETRI 부설연구소, freewill71@ensec.re.kr
논문번호 : KICS2012-10-510, 접수일자 : 2012년 10월 26일, 최종논문접수일자 : 2013년 4월 24일

경에서 큰 관심의 대상이 되는 기술이다.

예전부터 HE 스킴을 설계하기 위한 많은 노력들이 있어 왔다. 예를 들어, Goldwasser와 Micali[10], ElGamal[7], Paillier[12] 등의 암호화 스킴은 암호문의 덧셈 또는 곱셈 중의 하나에 대해서 준동형 계산이 (homomorphic evaluation) 가능하다. Boneh, Goh, Nissim는 한 번의 곱셈과 무제한의 덧셈에 대해서 준동형 계산이 가능한 스킴을 처음으로 제안하였다[2]. 그리고 2009년에 이르러서야 Gentry가 암호화된 데이터를 입력으로 하는 모든 함수의 준동형 계산이 가능한 FHE 스킴을 최초로 제시하였다[9].

Gentry의 FHE 설계 아이디어는 다음과 같다. 먼저 몇 번의 덧셈과 곱셈에 대해서만 준동형 계산이 가능한 SWHE(SomeWhat HE) 스킴을 설계하고, 부트스트래핑 개념을 도입한 후 부트스트래핑이 가능하면[1] leveled FHE로 확장 가능함을 증명한다. 이후에 부트스트래핑이 가능하도록 복호화 함수를 축소(squash)하는데 이로 인해 또 다른 문제의 안전성에도 의존하게 된다.

Gentry의 스킴에서 부트스트래핑을 이용하여 SWHE을 leveled FHE로 확장하는 과정은 계산량이 많고 더 많은 키를 필요로 하므로 구현 효율성 측면에서 매우 좋지 않다. 이에 다른 방법을 이용하여 이 과정을 대체하는 방법이 제시되었는데 Braskerski와 Vaikuntanathan[3], Braskerski, Gentry와 Vaikuntanathan[5]의 논문이 대표적이다. 이들의 논문은 모두 LWE(Learning With Errors) 문제에 기반하는데 암호문이 벡터 공간의 원소로서 암호문의 차원이 n인 두 암호문의 곱셈에 대해 준동형 계산을 하게 되면 차원이 n^2으로 커지게 된다. 따라서 암호문의 차원이 빨리 커지는 것을 막기 위하여 참고문헌 [5]에서는 키 전환(key switching)이란 과정을 도입하여 암호문의 차원을 다시 낮추도록 한다.(참고문헌 [3], [4]에서는 이 과정을 재선형화(re-linearlization)이라고 부름) 이 과정에 소용되는 연산량과 필요한 키가 상당히 커서 FHE 구현에 있어 상당한 부담이 된다.

본고에서는 키 전환 과정을 생략하고 직접 암호문의 차원을 낮추는 방식을 제안한다. 그 대신에 제안한 방식은 환(ring) 상의 이차방정식을 푸는 문제의 어려움이라는 새로운 안전성 가정을 필요로 하게 된다. 이 방식은 참고문헌 [3], [4], [5], [11] 모두에 적용되나 본고에서는 [5]를 기준으로 하여 설명한다.

1) Bootstrappable : Decrypt 함수 D와 NAND-augmented D 함수의 evaluate 가능

Ⅱ. 준동형 암호

2.1. 준동형 암호 정의

준동형 암호화 스킴 HE는 일단 비트 단위로 암호화하는 공개키 암호이고, HE = $\{KeyGen, Enc, Dec, Eval\}$로 구성된다. HE를 구성하는 각 함수에 대한 설명은 다음과 같다.

Keygen, Enc, Dec, Eval은 모두 PPT(Probabilistic Polynomial Time) 알고리즘이다.

- Key Generation : $(sk, pk) \leftarrow KeyGen(1^\lambda)$, λ는 security parameter
- Encryption : $c \leftarrow Enc_{pk}(\mu)$, $\mu \in 0,1$
- Decryption : $\mu^* \leftarrow Dec_{sk}(c)$
- Homomorphic Evaluation : $c_f \leftarrow Eval_{pk}(f, c_1, c_2, \cdots, c_t)$, f는 부울함수(boolean function)

HE가 $f \in C$ (함수 집합)에 대해

$$Dec_{sk}(Eval_{pk}(f, c_1, c_2, \cdots, c_t)) = f(\mu_1, \mu_2, \cdots, \mu_t)$$

을 만족하면 C-homomorphic이라고 한다. 또한 다항식 $s = s(\lambda)$가 존재하여 Eval(…)의 출력 길이가 s 비트 이하 즉 f 함수와 무관하면 콤팩트(compact)하다고 한다. 콤팩트하지 않은 Eval 함수 설계는 아주 쉽게 가능하므로 콤팩트 개념은 준동형 암호 설계시 매우 중요하다.

HE의 안전성은 의미론적인 안전성(semantic security)을 기준으로 한다. 즉 선택 평문 공격(CPA : Chosen Plaintext Attack)에 대해

$$(pk, Enc_{pk}(0)) \approx (pk, Enc_{pk}(1))$$

("$\approx$" 는 효율적인 알고리즘에 의해 구별 불가능함) 이면 안전하다고 말한다.

2.2. SWHE, FHE

어떤 스킴 HE가 콤팩트이고 차수가 낮은 부울함수에 대해 준동형 계산이 가능하면 SWHE(SomeWhat HE)이라고 한다. 어떤 스킴 HE가 콤팩트이고 모든 부울함수에 대해 준동형 계산이 가능하면 FHE이라고 한다. Keygen에 입력변수 1^L이 추가되어 $(sk, pk) \leftarrow KeyGen(1^\lambda, 1^L)$ depth[2]가 L인 부울함수에 대해 준

2) depth는 함수를 회로로 구현했을 때 그 회로의 depth를

동형 계산이 가능하고 $s(\lambda)$가 L에 무관하면 leveled FHE이라고 한다.

일반적으로 SWHE에서 준동형 계산을 여러 번 반복하게 되면 암호문의 노이즈(noise)가 증가하여 어느 순간 준동형(homomorphic) 성질이 없어진다. 따라서 SWHE를 FHE로 확장하려면 노이즈를 축소하는 방법을 필요로 한다. 이를 위해 일반적으로 부트스트래핑이나 모듈러스 전환 기법 등을 통해 준동형 계산을 한 암호문을 노이즈가 작은 암호문으로 대체하는 방법을 사용하게 된다.

2.3. 부트스트래핑 정리[9]

어떤 스킴 HE가 C-homomorphic이고 $\{Add-aug\ D,\ Mlt-aug\ D\}\subset C$ 이면, HE가 부트스트래핑이 가능하다고 정의한다. 여기에서

$$Add-aug\ D(sk,c,c') = D_{sk}(c)\oplus D_{sk}(c')$$

$$Mlt-aug\ D(sk,c,c') = D_{sk}(c)\ \bullet\ D_{sk}(c')$$

이다.

부트스트래핑 정리는 다음과 같다. 어떤 암호화 스킴이 부트스트래핑이 가능하면 leveled FHE 스킴으로 확장 가능하고, 부트스트래핑이 가능하고 weakly circular secure이면 FHE 스킴으로 확장 가능하다. 여기에서 암호 스킴이 weakly circular secure라 함은 모든 비밀키 비트를 암호화한 값 $Enc_{pk}(sk[i])$을 알고 있는 공격자에 대해 안전하다는 의미이다[9].

Ⅲ. BGV 스킴

편의상 참고문헌 [5]의 스킴을 BGV 스킴이라고 부르기로 한다. III절에서는 키 전환 과정을 포함하여 BGV 스킴을 자세하게 기술한다. 다만 SWHE을 FHE로 확장하기 위하여 필요한 노이즈 축소 기법인 모듈러스 전환과 부트스트래핑에 대해서는 본고에서 제안하는 내용과 직접적인 관련이 없으므로 기술을 생략한다.

3.1. 구성 방식

3.1.1. Leveled SWHE 구성

복호화 연산이 암호문과 비밀키의 내적(inner product) 계산 후 모듈로 연산으로 이루어진 LWE(Learning With Errors) 기반 공개키 암호 스킴을 구성한다. 암호문이 벡터이므로 덧셈의 계산은 XOR로 하면 내적 보존이 가능하므로 덧셈에 대해 준동형이 된다. 곱셈의 계산은 tensor product로 하고 비밀키도 tensor product하면 내적 보존이 가능하므로 곱셈에 대해 준동형이 된다. 다만 여기에서 암호문의 차원이 제곱으로 증가하므로 키 전환 과정을 통한 차원 축소를 한다. 이 과정에서 크기가 큰 개인키/공개키 쌍을 요구하므로 키 크기가 많이 증가하게 된다.

3.1.2. Leveled FHE 구성

곱셈의 준동형 계산을 하면 노이즈가 제곱이 되므로 노이즈 축소가 필요하다. Gentry의 부트스트래핑 대신에 새로운 노이즈 축소 기법 "모듈러스 전환"을 도입하여 정해진 모듈러스(modulus) q 대신에 레벨(level) 별로 다른 모듈러스를 사용한다. 처음 노이즈가 ϵ일 때 곱셈의 준동형 계산을 하면 ϵ^2이 되는데, 이를 같은 크기의 모듈러스로 scale-down하면 다시 크기가 ϵ이 되므로 이 과정을 반복한다. 이 방법을 통해 안전성이 LWE에만 의존하는 leveled FHE를 구성할 수 있다.

3.2. BGV 스킴

본 절에서는 BGV 스킴을 수식을 이용하여 표현한다. BGV 스킴은 일반적으로 LWE를 근간으로 하는데, RLWE(Ring LWE)를 근간으로 하면 구현 효율성이 증가하므로 본고에서는 RLWE 기반 BGV 스킴을 설명한다.

3.2.1. SWHE

먼저 $R_q = Z_q[x]/(x^d+1)$ (q는 홀수, d는 2의 멱승)라 하고 X는 R_q^m 상의 오류 분포(error distribution), 즉 norm이 작은 값으로 이루어진 벡터의 분포라고 하자. 그러면 SWHE는 다음과 같이 구성된다.

- Key Generation:
  - 개인키 sk 생성: $sk=(1,s)\in R_q^2$(s는 랜덤)
  - 공개키 pk 생성: $a\in R_q^m$ 선택, $e\in X$ 선택, $b=-sa+2e\in R_q^m$, $pk=M=\begin{pmatrix}b\\a\end{pmatrix}$
- Encryption: 평문 비트 $\mu\in 0,1$에 대해 랜덤 $r\in R_2^m(=R_2^{1\times m})$을 생성한 후 $c=Enc_{pk}(\mu)=rM^T+(\mu,0)=(c_0,c_1)\in R_q^2$ 계산
- Decryption:
  $Dec_{sk}(c)=[[<sk,c>]_q]_2=[[c_0+c_1s]_q]_2$
  ($[x]_q\in(-q/2,q/2)$: x를 q로 나눈 나머지)

의미

$(< sk, c >=< sk, rM^T + (\mu,0) >$
$=< sk, rM^T > + \mu$
$=< skM, r > + \mu$
$= 2 < e, r > + \mu$
이고 $[< e,r >]_q$가 작으므로
$Dec_{sk}(c) = [[< sk, c >]_q]_2 = \mu$)

- 덧셈에 대한 Homomorphic Evaluation: $Eval_{pk}(+, c, c^*) = c \oplus c^*$
- 곱셈에 대한 Homomorphic Evaluation: $Eval_{pk}(\bullet, c, c^*) = \vec{c} \otimes \vec{c^*}$

위에서 곱셈의 준동형 계산 과정을 거치고 난 암호문의 개인키는 sk가 아니라 $\vec{sk} \otimes \vec{sk}$가 된다. 즉

$Dec_{\vec{sk} \otimes \vec{sk}}(Eval_{pk}(\bullet, c, c^*)$
$= [[< vector(\vec{sk} \otimes \vec{sk}), vector(\vec{c} \otimes \vec{c^*}) >]_q]_2$
$= \mu \bullet \mu^*$

를 만족한다. 또한 곱셈의 준동형 계산 과정을 거친 새로운 암호문의 노이즈는 준동형 계산 전의 두 암호문의 노이즈의 곱과 같게 된다.

3.2.2. Key Switching

BGV SWHE에서 곱셈의 준동형 계산이 tensoring이므로 그 차원이 원 암호문의 차원의 곱셈이 된다. 따라서 c, c^*의 차원이 n이라면 $Eval_{pk}(\bullet, c, c^*)$의 차원은 n^2으로 커지게 된다. 이 차원의 증가가 기하급수적이기 때문에 BGV 스킴에서는 키 전환 과정을 통해 암호문의 차원을 다시 낮추게 된다. 이 암호문은 새로운 개인키/공개키 쌍에 대응되므로 이 과정을 키 전환이라고 부르는데 이 과정에서는 (sk, c)를

$Dec_{sk}(c) = [[< sk, c >]_q]_2 = [[< sk', c' >]_q]_2$
$= Dec_{sk'}(c')$

인 새로운 (sk', c')로 전환하게 된다.

1) c 작게 만들기

c를 $c = (c_0, c_1) = (\sum_{j=0}^{l} c_{0j} 2^j, \sum_{j=0}^{l} c_{1j} 2^j)$ ($l = \lceil \log_2 q \rceil$)와 같이 이진 전개한 후 $b_j = (c_{0j}, c_{1j})$라 하고 $c' = (b_0, b_1, \cdots, b_l)$, $sk' = (sk, 2sk, 2^2 sk, \cdots, 2^l sk) \bmod q$ 로 놓으면,

$$[< sk', c' >]_q = [\sum_{j=0}^{l} 2^j < sk, b_j >]_q$$
$$= [< sk, \sum_{j=0}^{l} 2^j b_j >]_q = [< sk, c >]_q$$

이므로 $Dec_{sk}(c) = Dec_{sk'}(c')$를 만족한다. 즉 c를 이진 전개한 새로운 c'으로 바꾸어 차원을 $2l$로 늘리는 대신 계수를 0 또는 1로 하여 작게 만들 수 있다. 대신에 대응하는 개인키도 sk'으로 변하게 된다.

2) 암호화하기

이제 다음은 (sk', c')을 기본 암호화 스킴을 이용하여 차원이 2인 새로운 (sk'', c'')로 대체하는 과정이다.

- Key Generation:
  - 개인키 sk'' 생성: $sk'' = (1, s'') \in R_q^2$($s''$는 랜덤)
  - 공개키 pk'' 생성: $a'' \in R_q^{2l}$ 선택, $e'' \in X$ 선택, $b'' = -s''a'' + 2e'' + sk' \in R_q^{2l}$,
    $pk'' = M = \begin{pmatrix} b'' \\ a'' \end{pmatrix}$
    (그러면
    $sk'' M = (1\ s'') \begin{pmatrix} b'' \\ a'' \end{pmatrix} = b'' + s''a'' = 2e'' + sk'$
    성립)
- c'' 생성: $c'' = c' M^T (\in R_q^2)$

c'', sk''은
$< sk'', c'' >=< sk'', c' M^T >=< sk'' M, c' >=$
$< 2e'' + sk', c' >= 2 < e'', c' > + < sk', c' >$
을 만족하고, $[< e'', c' >]_q$와 $[< sk', c' >]_q$가 작으므로
$[< sk'', c'' >]_q = 2[< e'', c' >]_q + [< sk', c' >]_q$
가 된다. 따라서
$Dec_{sk''}(c'') = [[< sk'', c'' >]_q]_2 = [[< sk', c' >]_q]_2$
$= Dec_{sk'}(c')$
을 만족한다.

이때 필요한 공개키는 R_q^{4l} 상의 원소이므로 그 크기가 $4l \lceil \log_2 q \rceil = 4(\lceil \log_2 q \rceil)^2$ 비트가 된다. 이는 원래 SWHE가 필요로 하던 공개키 크기인 $2m \lceil \log_2 q \rceil$ ($m = O(1)$[4] 또는 $m = O(\lceil \log_2 q \rceil)$[5])와 비교해 보면 상당한 큰 부담임을 알 수 있다.

Ⅳ. 키 전환 과정 대체하기

키 전환 과정을 위해서는 상당한 크기의 개인키/공개키 쌍을 생성해야 하는데다가 부가적인 연산을 요

구하여 FHE 구현 효율성 측면에서 큰 걸림돌이 된다. 이번 절에서는 환(ring) 상의 이차방정식을 푸는 문제의 어려움이라는 새로운 안전성 가정을 도입하고, 이를 기반으로 키 전환 과정을 생략하도록 하여 FHE 스킴의 구현 효율성을 제고할 수 있는 방법을 제안한다.

4.1. 제안 SWHE 방식

먼저 평문 비트 μ, μ^*에 대한 암호문 c, c^*에 대한 덧셈과 곱셈의 준동형 계산을 참고문헌 [4], [11]의 방법을 이용하여 다음과 같이 수행 가능하다. $c=(c_0, c_1)$, $c^*=(c_0{}^*, c_1{}^*)$라고 하면,

- $Eval_{pk}(+, c, c^*) = c \oplus c^*$
- $Eval_{pk}(\bullet, c, c^*) = (u, v, w)$,

$u = c_0 c_0{}^*$, $v = c_0 c_1{}^* + c_1 c_0{}^*$, $w = c_1 c_1{}^*$

위의 곱셈 준동형 계산 과정을 거치고 난 암호문의 개인키는 $sk=(1, s)$가 아니라 $sk'=(1, s, s^2)$가 된다. 즉

$$\begin{aligned} &Dec_{sk'}(Eval_{pk}(\bullet, c, c^*) \\ &= [[< (1, s, s^2), (u, v, w) >]_q]_2 \\ &= [[u + vs + ws^2]_q]_2 \\ &= [[c_0 + c_1 s]_q]_2 \bullet [[c_0{}^* + c_1{}^* s]_q]_2 \\ &= \mu \bullet \mu^* \end{aligned}$$

를 만족한다.

이때 다음과 같이 α, β를 생성한다. 임의의 랜덤 원소 $\alpha \in R_q$를 선택하고 $\beta = s^2 - \alpha s$를 R_q 상에서 계산하여 (α, β)를 공개키에 추가한다. 그리고 $Eval_{pk}(\bullet, c, c^*) = (u + \beta w, v + \alpha w)$으로 하고, ($u + \beta w$와 $v + \alpha w$는 모두 R_q 상의 연산을 통해 계산된 R_q 상의 원소) 이 암호문에 대응하는 개인키는 여전히 $sk=(1, s)$ 로 한다. 그러면

$$\begin{aligned} &Dec_{sk}(Eval_{pk}(\bullet, c, c^*) \\ &= [[< (1, s), (u + \beta w, v + \alpha w) >]_q]_2 \\ &= [[u + vs + ws^2]_q]_2 \\ &= \mu \bullet \mu^* \end{aligned}$$

을 만족하게 된다. 따라서 키 전환을 하지 않고도 곱셈의 준동형 계산을 수행한 후의 암호문의 차원이 2로 고정되도록 할 수 있다. 다만 여기에서도 공개키에 R_q 상의 원소 2개를 추가해야 하는데, 그 크기는 $2\lceil \log_2 q \rceil$ 비트로서 기존 키 전환 과정에서 필요한 공개키 크기에 비해 매우 작다. 매 곱셈을 할 때마다 위에서 제안한 곱셈에 대한 준동형 계산 과정을 통해 암호문은 항상 R_q 상의 2개의 원소로 표현되기 때문에 키 전환 과정이 필요 없게 된다.

이 SWHE를 기반으로 FHE 스킴으로 확장하는 것은 참고문헌 [4], [5]의 방법을 그대로 적용하면 가능하다. 즉 모듈러스 전환 기법을 적용하거나 기존의 부트스트래핑 정리를 이용하여 FHE 스킴으로 확장할 수 있다.

4.2. 안전성 분석

제안한 방식을 이용하여 곱셈의 준동형 계산을 하는 경우 RLWE 뿐만 아니라 다음과 같은 환 상의 이차방정식을 푸는 문제의 어려움에도 그 안전성을 의존하게 된다. 본 절에서는 이 문제의 안전성을 간단하게 분석해 본다.

- 문제: R_q 상의 원소 α, β에 대해 $s^2 + \alpha s + \beta = 0$의 해를 구하는 것이 어렵다.

위 문제는 q가 소수인 경우에는 쉽게 접근 가능하다. 즉 Z_q가 유한체가 되므로 $x^d + 1$을 Z_q 상에서 irreducible한 것들로 인수분해하면 R_q가 Z_q의 확장체의 곱(product)이 되고, 유한체 상에서는 이차방정식의 해를 쉽게 구할 수 있으므로 위 문제는 쉽게 해결 가능하다.

그러나 q가 소수가 아닌 경우에는 이러한 방법이 적용되지 않는다. 일반적인 방법으로는 s를 Z_q 상의 다항식으로 표현한 후 $s^2 + \alpha s + \beta$를 계산하고 여기에 나타나는 다항식의 모든 계수가 0이 되도록 하여 연립하여 풀면 된다. 즉 $s = \sum_{i=0}^{d-1} a_i x^i \, (a_i \in Z_q)$로 나타낸 후, $s^2 + \alpha s + \beta$를 계산하여 모든 x^j의 계수를 0으로 하면 a_i에 대한 다변수 2차 연립 방정식이 된다. 일반적으로 다변수 2차 연립 방정식의 해를 구하는 문제는 NP-hard로 알려져 있다[8]. 현재 이를 해결하는 알고리즘은 Gröbner 기저를 이용한 방법이 있는데, 참고문헌 [5], [11] 등에서 제안하는 크기인 $d = 2048$, $q \approx 2^{60}$ 정도의 크기를 이 방법으로 해결하기에는 아직 괴리가 크다[6].

따라서 제안 방식에서 파라미터 선택시 q는 반드시 소수가 아닌 홀수여야 한다.

V. 결 론

본고에서는 기존의 LWE 기반 FHE 스킴에서 준동형 계산 과정을 거친 암호문의 차원을 축소하기 위한 새로운 방법을 제안하였다. 이 방법은 기존의

키 전환 과정에 비해 추가되는 공개키 크기가 매우 작고 부가 연산량이 거의 없다는 측면에서 FHE 구현 효율성을 제고할 수 있다.

References

[1] K.-D. Chang and J.-L. Chen, “A survey of trust management in WSNs, internet of things and future internet,” *KSII Trans. Internet Inform. Syst.(TIIS)*, vol. 6, no. 1, pp. 5-23, 2012.

[2] D. Boneh, E.-J. Goh, and K. Nissim. “Evaluating 2-DNF formulas on ciphertexts,” in *Proc. Theory of Cryptography Conf. (TCC) ’05,* pp. 325-341, Cambridge, U.S.A., Feb. 2005.

[3] Z. Brakerski and V. Vaikuntanathan, “Efficient fully homomorphic encryption from (standard) LWE,” *IEEE Annu. Symp. Foundations Comput. Sci.,* pp. 22-25, Palm Springs, U.S.A., Oct. 2011.

[4] Z. Brakerski and V. Vaikuntanathan, “Fully homomorphic encryption from ring-LWE and security for key dependent messages,” in *Proc. CRYPTO 2011*, pp. 505-524, Santa Barbara, U.S.A., Aug. 2011.

[5] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, “Fully homomorphic encryption without bootstrapping,” in *Proc. Innovations in Theoretical Comput. Sci. (ITCS) 2012*, pp. 309-325, Porto, Portugal, July 2012.

[6] J. Ding, J. E. Gower, and D. S. Schmidt, *Multivariate Public Key Cryptosystems*, Springer, 2006.

[7] T. El-Gamal, “A public key cryptosystem and a signature scheme based on discrete logarithms,” in *Proc. CRYPTO 1984*, pp. 10-18, Santa Barbara, U.S.A., Aug. 1984.

[8] M. R. Garey and D. S. Johnson, *Computers and intractability: A guide to the theory of NP-completeness*, W. H. Freeman and Co., 1979.

[9] C. Gentry, “Fully homomorphic encryption using ideal lattices,” in *Proc. 41st ACM Symp. Theory of Computing (STOC) 2009*, pp. 169-178, Bethesda, U.S.A., May 2009.

[10] S. Goldwasser and S. Micali, “Probabilistic encryption and how to play mental poker keeping secret all partial information,” in *Proc. 14th ACM Symp. Theory of Computing (STOC) 1982*, pp. 365-377, San Francisco, U.S.A., May 1982.

[11] M. Naehrig, K. Lauter, and V. Vaikuntanathan, “Can homomorphic encryption be practical?,” in *Proc. ACM Cloud Computing Security Workshop (CCSW) 2011,* pp. 113-124, Chicago, U.S.A., Oct. 2011.

[12] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in *Proc. EUROCRYPT 1999*, pp. 223-238, Prague, Czech Republic, May 1999.

김 재 헌 (Jae-Heon Kim)
1991년 2월 KAIST 수학과 학사
1993년 2월 서울대학교 수학과 석사
2000년 8월 서울대학교 수학과 박사
2000년 4월~현재 ETRI 부설연구소
<관심분야> 통신보안, 암호, 정수론

유 상 경 (Sang-Kyung Yoo)
1999년 2월 KAIST 전기 및 전자공학과, 학사
2001년 2월 KAIST 정보통신공학과, 석사
2001년 1월~2003년 5월 : (주)텔리언 연구원
2003년 6월~현재 ETRI 부설연구소
2006년 Worcester Polytechnic Institute 방문연구원
<관심분야> 임베디드 시스템 보안, 암호

이 상 한 (Sang-Han Lee)
1993년 2월 경북대학교 전자공학과 학사
1997년 2월 경북대학교 전자공학과 석사
1997년 3월~2000년 2월 국방과학연구소
2000년 2월~현재 ETRI 부설연구소
<관심분야> 암호, 암호 가속 HW 구현