https://doi.org/10.33778/kcsa.2021.21.1.033

결합 준동형 인증 암호의 안전성 분석★

김 진 수*

요 약

클라우드 컴퓨팅 기술을 활용한 데이터의 아웃소싱은 제공된 데이터에 대한 제 3자 노출, 변조, 연산위임 결과의 신뢰성 등이 문제가 되고 있다. 이러한 보안 이슈들를 해결하기 위해 데이터를 암호화 한 후 연산 및 분석을 수행하는 준동형 암호가 큰 각광을 받고 있으며, 최근에는 준동형 암호에 인증 기능을 보강한 준동형 인증 암호들이 제안되었다. 이 암호를 활용하면 정보의 유출과 민감한 개인정보에 대한 침해 문제없이 데이터의 분석이 가능함과 동시에 위임된 연산에 대한 신뢰성을 보장받을 수 있기 때문이다. 그러나 준동형 인증 암호 설계에 대한 연구는 아직 초기 단계이다. 제시된 준동형 인증 암호들 중 실용적인 스킴들은 그 안전성이 매우 낮거나, 반대로 안전성이 높은 경우에는 실용적이지 못하다. 또한 준동형 메시지 인증 스킴과 준동형 암호를 바탕으로 준동형 인증 암호를 설계하는 기법에 대한 분석이 부재한 실정이다. 본 논문에서는 준동형 메시지 인증 스킴과 준동형 암호를 결합하여 설계하는 기법에 대한 안전성 분석하였다. 분석 결과 위조 불가능한 준동형 메시지 인증 스킴을 이용하여 준동형 인증 암호를 구성하면 준동형 인증 암호 역시 위조불가능성을 갖지만 강한 위조 불가능성의 경우는 그렇지 않았다. 한편 구별불가능성을 갖는 준동형 암호를 이용하여 준동형 인증 암호를 설계하더라도 구별불가능성에 대한 안전성을 만족하지 못함을 확인하였다.

Analysis of Homomorphic Authenticated Encryption (Encrypt with Authenticate Construction)

Jinsu Kim*

ABSTRACT

Data outsourcing utilizing the Cloud faces a problem of the third-party exposure, modulation, and reliability for the provided computational delegation results. In order to solve those problematic security issues, homomorphic encryption(HE) which executes calculation and analysis on encrypted data becomes popular. By extension, a new type of HE with a authentication functionality, homomorphic authenticated encryption(HAE) is suggested. However, a research on the HAE is on the initial stage. Furthermore, based on a message authenticated scheme with HE, the method and analysis to design is still absent. This paper aims to analyze an HAE, with a generic combination of a message authenticated scheme and a HE, known as “Encrypt with Authentication”. Following a series of analysis, we show that by adopting a unforgeable message authenticated scheme, the generically constructed HAE demonstrated an unforgeability as well. Though, a strong unforgeability is not the case. This paper concludes that although indistinguishable HE can be applied to design the HAE, a security issue on the possibility of indistinguishability is still not satisfied.

Key words : HAE, homomorphic authenticated encryption, unforgeability, indistinguishability

접수일(2021년 02월 28일), 게재확정일(2021년 03월 18일)

★ 이 논문은 2021년 해군사관학교 해양연구소 학술연구과제 연구비의 지원으로 수행된 연구임.

* 해군사관학교/수학과(주저자)

1. 서 론

클라우드 컴퓨팅(cloud computing) 기술과 관련 서비스의 발전에 힘입어 정보 분석을 위한 대용량의 데이터(data)의 아웃소싱(outsourcing)과 연산 위임(delegation)이 보다 활발해지고 있으며, 매우 중요한 기술로 자리 잡아 가고 있다. 소규모 창업에서부터 대기업에 이르기까지 데이터를 기반 사업체들이 생산하는 데이터의 양은 무어의 법칙(Moore's Law)보다 빠르게 증가하고 있으며, 이러한 빅 데이터(big data) 시대에는 대용량의 데이터를 관리하거나 분석하여 유용한 정보를 추출하는 것이 매우 중요한 과제이다. 한편 업체에서는 이를 위한 많은 노력과 비용이 요구되므로 클라우드 컴퓨팅 기술을 이용하여 비용을 절약하고, 분석효율을 높이고 있다. 그러나 이러한 데이터 아웃소싱은 위임된 데이터의 관리 면에서 여러 가지 보안 이슈를 발생시킨다. 서비스 업체로 위임한 데이터의 변조, 유출 문제뿐만 아니라 위임한 데이터에 대한 분석 결과를 항상 신뢰해야하는 문제도 발생한다. 이러한 문제들은 데이터 아웃소싱을 위해서 반드시 해결되어야 하지만 그 대책은 완벽하지 못한 실정이다. 대표적인 예로 빅 데이터 처리 플랫폼(platform) 하둡(Hadoop)은 커버로스(Kerberos)라는 인증 프로토콜(authentication protocol)을 사용하지만 이 프로토콜에는 암호화 기능이 없어 여전히 데이터 정보의 유출이 발생 할 수 있다 [1-2].

대두된 보안 문제를 해결하기 위해 사용하는 방법 중 하나는 변조 방지 하드웨어(tamper-resistant hardware)를 사용하여 비인가자의 접근을 통제하는 것이다. 또한 중요한 데이터에 대해서는 상용 빅 데이터 보안 툴(tool)을 이용, 암호화하여 클라우드 서버에 저장함으로써 정보 유출을 막을 수 있다. 그러나 이러한 방법들은 데이터를 이용하여 필요 정보를 탐색하거나 데이터에 대한 분석을 수행 할 때 문제가 발생한다. 이러한 작업을 위해서는 복호화 과정이 반드시 선행이 되어야 하므로 결국 복호화-분석-재암호화 과정으로 이어져 분석 과정 중 주요 정보의 변조 및 유출 문제가 여전히 발생할 수 있으며, 또한 그 과정이 매우 비효율적이다.

이러한 배경에서 학계에서는 다양한 암호학적 해결 방법들이 연구되어 왔다. 보안 정보 검색(Private Information Retrieval) [3-5], 탐색 가능 암호(Searchable Encryption) [6-10]를 이용하면 신뢰할 수 없는 서비스 제공자(service provider)로부터 앞서 언급한 데이터에 대한 유출과 비효율적인 복호화 선행과정 없이 키워드(keyword), 순위(rank), 부분집합 정보 등이 검색 가능하다. 한편 준동형 암호(homomorphic encryption) [11-14]를 이용하면 암호화된 데이터 상태에서 평문에 대한 연산을 수행할 수 있기 때문에 복호화 과정 없이 검색 및 분석을 바탕으로 정보의 재가공이 가능하다. 여기서도 여전히 보안에 대한 이슈는 남아있다. 준동형 암호는 복호화 과정 없이 서비스 제공자가 암호화된 데이터에 대해 연산이 가능하게 하지만 그 연산 결과에 대한 신뢰성은 보장하지 못한다. 연산의 주체가 정해진 암호문이 아닌 다른 임의의 암호문으로 연산을 수행하거나 제 3자가 임의의 암호문을 생성하여 클라우드 서버의 연산결과와 임의로 조작할 수도 있기 때문이다. 이에 대한 해결책으로 암호화와 동시에 인증 기능을 가능하게 만드는, 즉 인가된 사용자만 암호문을 생성할 수 있는 인증 암호(authenticated encryption) [15-16]들이 제안되어 왔으며, 준동형 암호의 보완이자 아웃소싱에 대한 암호학적 최종 해결책으로 준동형 인증 암호(homomorphic authenticated encryption)가 비로소 최근에 제안되었다 [17-19].

준동형 인증 암호에 대한 연구는 아직 초기 단계로써 [17]에서 제안한 준동형 인증 암호는 높은 안전성을 갖는 반면 실용적이지 못하며, [18], [19]에서 제안된 암호는 실용적이나 그 안전성 매우 낮은 수준이다. 한편 준동형 인증 암호의 설계 아이디어는 준동형 암호(homomorphic secret encryption)와 준동형 메시지 인증(homomorphic message authentication)을 합성하는 방식에 기반하는데, 아직 그에 대한 분석이 이루어지지 않고 있다. 본 논문에서는 일반적으로 알려진 세 가지 합성 방식([20]) 중 가장 효율적인 방식인 결합 준동형 인증 암호(encrypt with authenticate construction) 설계 방식에 대해 안전성 분석을 수행하였다. 분석 결과 위조 불가능한 준동형 메시지 인증 스킴을 이용하여 준동형 인증 암호를 구성하면 준동형 인증 암호 역시 위조 불가능성을 갖지만 강한 위조

https://doi.org/10.33778/kcsa.2021.21.1.033

불가능성의 경우는 그렇지 않았다. 한편 구별불가능성을 갖는 준동형 암호를 이용하여 준동형 인증 암호를 설계하더라도 구별불가능성에 대한 안전성을 만족하지 못함을 확인하였다. 2장에서는 관련 연구 및 안전성 분석 수행에 필요한 관련 용어들을 정리하여 제시하였고, 3장에서는 2장을 배경으로 분석 결과를 마지막 4장에는 결론 및 향후 연구방향을 제시하였다.

2. 관련 연구 및 정의

최초의 준동형 인증 암호는 주치홍, 윤아람에 의해 제안되었다.([17]) 제안된 암호는 근사 최대공약수 문제(error-free approximate GCD, EF-AGCD)를 이용하여 선택 암호문 공격자에 대한 구별 불가능성(IND-CCA)과 강한 위조 불가능성(SUF-CCA)을 갖도록 설계되었으며, 암호화 및 복호화 과정이 비교적 단순하다. 반면 기반 문제의 어려움 가정과 암호문간의 곱셈 연산 지원을 위한 파라미터(parameter) 크기(size)의 증가로 인해 안전성을 보장하기 위해서는 준동형 인증 암호 알고리즘 전체의 크기가 커지는 단점이 있다. 이러한 단점을 극복함과 동시에 공개키 방식의 준동형 인증 암호가 Struck 등에 의해 제안되었다.([19]) 제안된 암호는 선택 암호문 공격자에 대한 구별불가능성과 위조 불가능성을 만족하며, 공개키 방식임에도 효율성을 달성하기 위하여 [17]과 달리 암호문에 대한 덧셈 연산만을 지원한다. 이후 대칭키 방식으로 덧셈 연산만을 지원하는 가장 효율적인 방식의 준동형 인증 암호가 제안되었다.([18]) 반면 선택 평문 공격자에 대한 구별 불가능성(IND-CPA)과 위조 불가능성만을 만족하기 때문에 다른 준동형 인증 암호와 비교해 보았을 때 상대적으로 안전성 수준이 낮음을 알 수 있다.

이 세 가지 준동형 인증 암호들은 모두 인증에 사용하는 라벨(label)을 랜덤화(randomization)하여 평문 노출을 방지하는데 사용한다. 이러한 설계방식은 [20]에 제안된 인증 암호에 대한 일반적인(generic) 설계 방식 중 인증 후 암호화(MAC-then-encrypt) 방식에 기인하며, 따라서 알려진 일반적인 설계 방식, 인증 및 암호 결합(MAC-and-encrypt), 암호화 후 인증(encrypt-then-MAC), 인증 후 암호화의 준동형 버전(version) 대한 연구가 필수적이다. [20]에서는 준동형 기능이 없는 메지지 인증 스킴과 대칭키 암호를 이용하여 인증 암호를 설계 및 분석하였으므로, 암호문간의 연산이 가능한 경우에 대한 새로운 정의와 분석이 필요하다.

결합 준동형 인증 암호의 안전성 분석을 위하여 준동형 메시지 인증 스킴과 준동형 인증 암호 관련 용어들을 정의한다. 표기 방식은 [17]을 따라 정의하였다.

2.1 준동형 메시지 인증 스킴과 그 안전성

준동형 메시지 인증 스킴 HMA(Homomorphic Message Authentication)은 다음과 같은 네 개의 확률적 다항시간 알고리즘(probabilistic polynomial-time algorithm) 키 생성(KeyGen), 인증(Aut), 연산(Eval), 확인(Ver)으로 구성된다.

HMA

- $KeyGen(1^{\lambda})$: 안전성 파라미터 λ에 대하여 연산키 ek와, 비밀키 sk를 출력한다.
- $Aut(\tau, m, sk)$: 입력 라벨(label) $\tau \in L$, 메시지 $m \in M$, 비밀키 sk를 이용하여, 인증 μ를 생성 및 출력한다.
- $Eval(f, \mu_1, \mu_2, \cdots, \mu_l, ek)$: 각 라벨 $\tau_i \in L$ $(1 \le i \le l)$ 와 메시지 $m_i \in M (1 \le i \le l)$ 에 대한 인증 μ_i와 함수 $f : M^l \rightarrow M$에 대하여, 연산키 ek로 인증 연산 $\mu = f(\mu_1, \mu_2, \cdots, \mu_l)$ 를 계산 및출력한다.
- $Ver((f, \tau_1, \tau_2, \cdots, \tau_l), m, \mu, sk)$: 라벨 프로그램 $(f, \tau_1, \tau_2, \cdots, \tau_l)$과 메시지 m에 대하여, 비밀키 sk를 이용하여 인증 μ에 대한 검증결과 $h \in \{0,1\}$를 계산 및 출력한다.

여기서 $Eval$, Ver는 결정론적(deterministic) 알고리즘이며, 연산키 ek는 $Eval$ 알고리즘 연산에 필요한 평문과 라벨 정보 등을 포함한다. 함수 f는 입력 값에 따라 달라지지만 동일한 연산을 수행하므로 문맥에 따라 해석한다. (예를 들면, 라벨 프로그램

$(f, \tau_1, \tau_2, \cdots, \tau_l)$의 경우 $f : M^l \rightarrow M$이 아닌 $f : L^l \rightarrow L$이다.)

준동형 메시지 인증 스킴에 대한 안전성은 공격자가 임의로 비밀정보 없이 위조(forgery)를 생성해 낼 수 있는지 없는 지로 결정된다. 이때 공격자의 능력 정도에 따라 선택 평문 공격(chosen message attack)과 선택 인증 공격(chosen tag attack) 두 가지 공격으로 나누어지며([17]), 선택 평문 공격의 경우 공격자는 임의의 평문에 대한 인증을 질의(query)하여 얻을 수 있는 반면, 선택 인증 공격의 경우에는 평문에 대한 인증 질의에 더하여 인증에 대한 확인 질의 또한 가능하다. 다만 두 가지 경우 모두 공격자의 인증 질의에 대한 결과를 생성하기 위해 이용되는 라벨은 재활용되지 않고 반드시 한번만 사용되어야 한다. 따라서 인증 오라클(oracle)은 다음과 같은 인증 질의에 대한 기록(history)을 유지함으로써 공격자의 인증 질의에 대하여 라벨의 재사용을 방지한다.

정의 1. [인증 기록] 인증 기록은 함수 $H : L \rightarrow \{\perp\} \cup (M \times W)$ 로써 다음과 같이 공격자의 인증 질의에 따라 정의된다.

- 처음 H의 함수 값은 모두 $\perp$이다.
- 공격자의 인증 질의 (τ, m)에 대하여 만일 $H(\tau) = \perp$이면, $\mu = \mathrm{Aut}(\tau, m, sk)$를 계산하여 공격자에게 알려주고 $H(\tau) := (m, \mu)$ 로 업데이트 한다. $H(\tau) \neq \perp$이면 인증 질의에 대한 답변은 거절한다.

메시지 인증 스킴에 대한 공격자의 목표인 위조는 다음과 같이 두 가지 형태로 정의한다.([17]) 위조란 비밀키를 이용하여 생성한 인증 및 그 인증의 합 또는 곱으로 재생성된 인증이외에 비밀키 정보없이 준동형 인증 스킴의 확인과정(Ver)을 통과하는 인증들을 생성하는 행위를 의미한다. 공격자의 위조 시도 $\big((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu'\big)$ 는 다음 두 가지 조건 중 하나를 만족하면 위조라 한다.

1) $Ver\big((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu', sk\big) = 1$,
그리고 $f\big((m_i)_{i \in I}\big)$는 비-상수함수

2) $Ver\big((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu', sk\big) = 1$,
그리고 $f\big((m_i)_{i \in I}\big)$는 상수함수,
$f\big((m_i)_{i \in I}\big) \neq m'$

여기서 $I \subset \{1, 2, \cdots, l\}$ 는 라벨에 사용된 인덱스 집합이며, $f\big((m_i)_{i \in I}\big)$는 정의역이 $M^{l - |I|}$, 공역이 M인 함수이다.

강한 위조 역시 비밀키를 이용하여 생성한 인증 및 그 인증의 합 또는 곱으로 재생성된 인증이외에 비밀키 정보없이 준동형 인증 스킴의 확인과정을 통과하는 인증들을 생성하는 행위를 의미한다. 위조와 비교하자면 강한 위조는 정의상 모든 위조를 포함하므로 공격자의 달성목표 측면에서는 위조에 비해 강한 위조가 상대적으로 낮은 목표가 된다. 공격자의 강한 위조 시도 $\big((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu'\big)$ 는 다음 두 가지 조건 중 하나를 만족하면 강한 위조라 한다.

1) $Ver\big((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu', sk\big) = 1$,
$f\big((m_i)_{i \in I}\big)$ 또는 $Eval\big(f, \mu_1, \mu_2, \cdots, \mu_l, ek\big)$
는 비상수함수
2) $Ver\big((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu', sk\big) = 1$,
$f\big((m_i)_{i \in I}\big)$와 $Eval\big(f, \mu_1, \mu_2, \cdots, \mu_l, ek\big)$는
상수함수, $f\big((m_i)_{i \in I}\big) \neq m'$ 또는
$Eval\big(f, \mu_1, \mu_2, \cdots, \mu_l, ek\big) \neq \mu'$

여기서도 $I \subset \{1, 2, \cdots, l\}$ 는 라벨에 사용된 인덱스 집합이며, $Eval\big(f, \mu_1, \mu_2, \cdots, \mu_l, ek\big)$ 는 정의역이 $W^{l - |I|}$, 공역이 W인 함수이다.

위조에 대한 정의를 바탕으로 공격자에 대한 위조 불가능성을 다음과 같은 게임으로서 정의한다. 우선 선택 평문 공격자에 대한 위조 불가능성은 다음과 같은 게임을 실행하여 공격자의 위조 생성 성공 확률이 무시할 만큼 작을 때(negligible) 위조 불가능하다고 정의한다.

$UF-CMA_A^{\lambda}$ (선택 평문 공격자 A의 위조 게임)

https://doi.org/10.33778/kcsa.2021.21.1.033

- 초기화(Initialization, Ini) : 첼린저(challenger)는 안전성 파라미터 λ에 대한 연산키와, 비밀키 ek, sk를 $KeyGen(1^{\lambda})$를 이용하여 발급받고, 연산키 ek를 공격자 A에게 넘겨준다.
- 질의(Queries, Que) : 공격자 A의 (τ, m)에 대한 인증 질의에 대해 첼린저는 $\mu \leftarrow Aut(\tau, m, sk)$를 계산하여, 공격자 A에게 μ로써 질의에 응답한다.
- 최종(Finalization, Fin) : 공격자 A는 위조 시도 $\left((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu'\right)$를 첼린저에게 제출한다. .첼린저는 위조여부를 확인하고, 위조이면 1, 위조가 아니면 0을 출력하고 게임을 종료한다.

위 게임에서 공격자 A의 이득은

$Adv_{HMA, A}^{UF-CMA}(\lambda) := \Pr\left[UF-CMA_A^{\lambda} = 1\right]$

로 정의하며, 그 값이 임의의 확률적 다항시간 공격자에 대하여 무시할 만큼 작으면 준동형 메시지 인증 스킴 HMA는 선택 평문 공격자에 대한 위조 불가능성($UF-CMA$)을 갖는다고 한다. 선택 평문 공격자에 대한 강한 위조 불가능성은 최종 단계에서 강한 위조 인지의 여부를 확인하며, 나머지는 동일하게 게임을 진행한다. 마찬가지로 공격자의 이득은

$Adv_{HMA, A}^{SUF-CMA}(\lambda) := \Pr\left[SUF-CMA_A^{\lambda} = 1\right]$

로 정의하며, 그 값이 임의의 확률적 다항시간 공격자에 대하여, 무시할 만큼 작으면 준동형 메시지 인증 스킴 HMA는 선택 평문 공격자에 대한 강한 위조 불가능성($SUF-CMA$)을 갖는다고 한다.

선택 인증 공격자에 대한 위조 불가능성은 다음과 같은 게임을 실행하여 공격자의 위조 생성의 성공 확률이 무시할 만큼 작을 때 위조 불가능하다고 정의한다.

$UF-CTA_A^{\lambda}$(선택 인증 공격자 A의 위조 게임)

- 초기화(Initialization, Ini) : 첼린저는 안전성 파라미터 λ에 대한 연산키와, 비밀키 ek, sk를 $KeyGen(1^{\lambda})$ 를 이용하여 발급받고, 연산키 ek를 공격자 A에게 넘겨준다.
- 질의(Queries, Que) : 공격자 A의 (τ, m)에 대한 인증 질의에 대해 첼린저는 $\mu \leftarrow Aut(\tau, m, sk)$를 계산하여, 공격자 A에게 μ로써 질의에 응답한다. 한편, 공격자 A의 $\left((f, \tau_1, \tau_2, \cdots, \tau_l), m, \mu\right)$에 대한 검증 질의에 대해 첼린저는 $h \leftarrow Ver\left((f, \tau_1, \tau_2, \cdots, \tau_l), m, \mu, sk\right)$를 계산하여, 공격자 A에게 h로써 질의에 응답한다.
- 최종(Finalization, Fin) : 공격자 A는 위조 시도 $\left((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu'\right)$를 첼린저에게 제출한다. 첼린저는 위조 여부를 확인하고, 위조이면 1, 위조가 아니면 0을 출력하고 게임을 종료한다.

위 게임에서 공격자 A의 이득은

$Adv_{HMA, A}^{UF-CTA}(\lambda) := \Pr\left[UF-CTA_A^{\lambda} = 1\right]$로 정의하며, 그 값이 임의의 확률적 다항시간 공격자에 대하여 무시할 만큼 작으면 준동형 메시지 인증 스킴 HMA는 선택 인증 공격자에 대한 위조 불가능성($UF-CTA$)을 갖는다고 한다. 선택 인증 공격자에 대한 강한 위조 불가능성은 마찬가지로 최종 단계에서 강한 위조 인지의 여부를 확인하며 나머지는 동일하게 게임을 진행한다. 마찬가지로 공격자의 이득은

$Adv_{HMA, A}^{SUF-CTA}(\lambda) := \Pr\left[SUF-CTA_A^{\lambda} = 1\right]$로 정의하며 그 값이 임의의 확률적 다항시간 공격자에 대하여 무시할 만큼 작으면 준동형 메시지 인증 스킴 HMA는 선택 인증 공격자에 대한 강한 위조 불가능성($SUF-CTA$)을 갖는다고 한다.

이상의 정의를 종합해보면 위조보다는 강한 위조를 달성하기 쉽고, 선택 평문 공격자 보다는 선택 인증 공격자가 보다 강한 능력을 갖고 있음을 알 수 있다. 따라서 선택 인증 공격자에 대한 (강한) 위조 불가능성을 갖춘 준동형 메시지 인증 스킴은 선택 평문 공격자에 대한 (강한) 위조 불가능성을 만족하며, 공격자의 능력에 관계없이 강한 위조 불가능성을 갖춘 준동형 메시지 인증 스킴은 위조 불가능성을 갖는다.

2.2 준동형 인증 암호와 그 안전성

준동형 인증 암호 HAE(Homomorphic Authenticated -Encryption)는 다음과 같은 네 개의 하위 알고리즘 키생성(KeyGen), 암호화(Enc), 연산(Eval), 복호화(Dec)로 구성된다.

HAE

- $KeyGen(1^{\lambda})$: 안전성 파라미터 λ에 대하여 연산키 ek와 비밀키 sk를 생성 및 출력한다.
- $Enc(\tau, m, sk)$: 입력 라벨 $\tau \in L$, 메시지 $m \in M$에 대하여, 비밀키 sk를 이용하여 암호문 $c \in C$를 출력한다.
- Eval$(f, c_1, c_2, \cdots, c_l, ek)$: 각 메시지 $m_i \in M$에 대한 암호문 c_i와 함수 $f : M^l \rightarrow M$에 대하여 연산키 ek를 이용하여, $c = f(c_1, c_2, \cdots, c_l)$를 출력한다.
- Dec$(\left(f, \tau_1, \tau_2, \cdots, \tau_l\right), c, sk)$: 라벨 프로그램 $\left(f, \tau_1, \tau_2, \cdots, \tau_l\right)$과 암호문 c에 대하여 비밀키 sk를 이용하여 복호화 결과로 $f\left(\left(m_i\right)_{i \in I}\right)$ 또는 $\perp$을 출력한다.

여기서 $Eval$, Dec는 결정론적(deterministic) 알고리즘이며, 연산키 ek는 연산에 필요한 평문과 라벨 정도 등을 포함한다. 준동형 인증 암호 역시 공격자의 인증 질의에 대한 결과를 생성하기 위해 이용되는 라벨은 재활용되지 않고 반드시 한번만 사용되어야 한다. 따라서 암호화 오라클은 다음과 같은 암호화 질의에 대한 기록을 유지함으로써 공격자의 암호화 질의에 대하여 라벨의 재사용을 방지한다.

정의 2. [인증 기록] 인증 기록은 함수 $H : L \rightarrow \{\perp\} \cup (M \times C)$ 로써 다음과 같이 공격자의 인증 질의에 따라 그 함수가 정의된다.

- 처음 H의 함수 값은 모두 $\perp$이다.
- 공격자의 암호화 질의 (τ, c)에 대하여 만일 $H(\tau) = \perp$이면, $c = \text{Enc}(\tau, m, sk)$를 계산하여 공격자에게 알려주고 $H(\tau) := (m, c)$로 업데이트 한다. $H(\tau) \neq \perp$이면 인증 질의에 대한 답변은 거절한다.

준동형 인증 암호에 대한 공격자의 목표는 암호문의 해독과 위조 두 가지이며, 암호문 해독에 대한 안전성 모델은 준동형 암호와 유사하게 정의된다. 위조는 준동형 메시지 인증 스킴과 유사한 방식으로 다음과 같이 두 가지 형태로 정의한다.([17])

우선 위조란 비밀키를 이용하여 생성한 암호문 및 그 암호문간의 합 또는 곱으로 재생성된 암호문이외에 비밀키 정보 없이 준동형 인증 암호의 복호화 과정을 통과하는 암호문들을 생성함을 의미한다. 공격자의 위조 시도 $\left(\left(f, \tau_1, \tau_2, \cdots, \tau_l\right), c'\right)$ 는 다음 두 가지 조건 중 하나를 만족하면 위조라 한다.

1) $Dec\left(\left(f, \tau_1, \tau_2, \cdots, \tau_l\right), c', sk\right) \neq \perp$,
   $f\left(\left(m_i\right)_{i \in I}\right)$는 비-상수함수
2) $Dec\left(\left(f, \tau_1, \tau_2, \cdots, \tau_l\right), c', sk\right) \neq \perp$,
   $f\left(\left(m_i\right)_{i \in I}\right)$는 상수함수,
   $f\left(\left(m_i\right)_{i \in I}\right) \neq Dec\left(\left(f, \tau_1, \tau_2, \cdots, \tau_l\right), c', sk\right)$

여기서 $I \subset \{1, 2, \cdots, l\}$는 라벨에 사용된 인덱스 집합이며, $f\left(\left(m_i\right)_{i \in I}\right)$는 정의역이 $M^{l - |I|}$, 공역이 M인 함수이다.

공격자의 강한 위조 시도 $\left(\left(f, \tau_1, \tau_2, \cdots, \tau_l\right), c'\right)$는 다음 두 가지 조건 중 하나를 만족하면 강한 위조라 한다.

1) $Dec\left(\left(f, \tau_1, \tau_2, \cdots, \tau_l\right), c', sk\right) \neq \perp$,
   $f\left(\left(m_i\right)_{i \in I}\right)$ 또는 $Eval\left(f, c_1, c_2, \cdots, c_l, ek\right)$
   는 비-상수함수
2) $Dec\left(\left(f, \tau_1, \tau_2, \cdots, \tau_l\right), c', sk\right) \neq \perp$,
   $f\left(\left(m_i\right)_{i \in I}\right)$와 $Eval\left(f, c_1, c_2, \cdots, c_l, ek\right)$는
   상수함수,
   $f\left(\left(m_i\right)_{i \in I}\right) \neq Dec\left(\left(f, \tau_1, \tau_2, \cdots, \tau_l\right), c', sk\right)$
   또는 $Eval\left(f, c_1, c_2, \cdots, c_l, ek\right) \neq c'$

여기서 $I \subset \{1, 2, \cdots, l\}$는 라벨에 사용된 인덱스 집합이며, $Eval\left(f, c_1, c_2, \cdots, c_l, ek\right)$ 는 정의역이 $C^{l - |I|}$, 공역이 C인 함수이다.

선택 평문 공격자에 대한 위조 불가능성은 다음과 같은 게임을 실행하여 공격자의 위조 생성의 성공 확률이 무시할 만큼 작을 때 위조 불가능하다고 정의한다.

https://doi.org/10.33778/kcsa.2021.21.1.033

$UF-CPA_A^{\lambda}$(공격자 A의 위조 게임)

- 초기화(Initialization, Ini) : 첼린저는 안전성 파라미터 λ에 대한 연산키와, 비밀키 ek, sk를 $KeyGen(1^{\lambda})$를 이용하여 발급받고, 연산키 ek를 공격자 A에게 넘겨준다.
- 질의(Queries, Que) : 공격자 A의 (τ, m)에 대한 암호화 질의에 대해 첼린저는 $Enc(\tau, m, sk)$를 계산하여, 공격자 A의 질의에 응답한다.
- 최종(Finalization, Fin) : 공격자 A는 위조 시도 $\left((f, \tau_1, \tau_2, \cdots, \tau_l), c'\right)$를 첼린저에게 제출한다. 첼린저는 위조여부를 확인하고, 위조이면 1, 위조가 아니면 0을 출력하고 게임을 종료한다.

위 게임에서 공격자 A의 이득은 $Adv_{HAE, A}^{UF-CPA}(\lambda) := \Pr\left[UF-CPA_A^{\lambda} = 1\right]$로 정의하며 그 값이 임의의 확률적 다항시간 공격자에 대하여 무시할 만큼 작으면 준동형 인증 암호 HAE는 선택 평문 공격자에 대한 위조 불가능성($UF-CPA$)을 갖는다고 한다. 선택 평문 공격자에 대한 강한 위조 불가능성은 최종 단계에서 강한 위조 인지의 여부를 확인하며 나머지는 동일하게 게임을 진행한다.

마찬가지로 공격자의 이득은 $Adv_{HAE, A}^{SUF-CPA}(\lambda) := \Pr\left[SUF-CPA_A^{\lambda} = 1\right]$로 정의하며 그 값이 임의의 확률적 다항시간 공격자에 대하여 무시할 만큼 작으면 준동형 인증 암호 HAE는 선택 평문 공격자에 대한 강한 위조 불가능성($SUF-CPA$)을 갖는다고 한다.

선택 암호문 공격자에 대한 위조 불가능성은 다음과 같은 게임을 실행하여 공격자의 위조 생성의 성공 확률이 무시할 만큼 작을 때 위조 불가능하다고 정의한다.

$UF-CCA_A^{\lambda}$(선택 암호문 공격자 A의 위조 게임)

- 초기화(Initialization, Ini) : 첼린저는 안전성 파라미터 λ에 대한 연산키와, 비밀키 ek, sk를 $KeyGen(1^{\lambda})$를 이용하여 발급받고, 연산키 ek를 공격자 A에게 넘겨준다.
- 질의(Queries, Que) : 공격자 A의 (τ, m)에 대한 암호화 질의에 대해 첼린저는 $Enc(\tau, m, sk)$를 계산하여, 공격자 A에게 c로써 질의에 응답한다. 복호화 질의 $\left((f, \tau_1, \tau_2, \cdots, \tau_l), c\right)$에 대해서는 $Dec\left((f, \tau_1, \tau_2, \cdots, \tau_l), c, sk\right)$로 질의에 응답한다.
- 최종(Finalization, Fin) : 공격자 A는 위조 시도 $\left((f, \tau_1, \tau_2, \cdots, \tau_l), c'\right)$를 첼린저에게 제출한다. 첼린저는 위조여부를 확인하고, 위조이면 1, 위조가 아니면 0을 출력하고 게임을 종료한다.

위 게임에서 공격자 A의 이득은 $Adv_{HAE, A}^{UF-CCA}(\lambda) := \Pr\left[UF-CCA_A^{\lambda} = 1\right]$로 정의하며 그 값이 임의의 확률적 다항시간 공격자에 대하여 무시할 만큼 작으면 준동형 인증 암호 HAE는 선택 암호문 공격자에 대한 위조 불가능성($UF-CCA$)을 갖는다고 한다. 선택 인증 공격자에 대한 강한 위조 불가능성은 최종 단계에서 강한 위조 인지의 여부를 확인하며 나머지는 동일하게 게임을 진행한다.

마찬가지로 공격자의 이득은 $Adv_{HAE, A}^{SUF-CCA}(\lambda) := \Pr\left[SUF-CCA_A^{\lambda} = 1\right]$로 정의하며, 그 값이 임의의 확률적 다항시간 공격자에 대하여 무시할 만큼 작으면 준동형 인증 암호 HAE는 선택 암호문 공격자에 대한 강한 위조 불가능성($SUF-CCA$)을 갖는다고 한다.

3. 결합 준동형 암호의 안전성 분석

준동형 인증 암호를 설계할 수 있는 일반적인 방법 중 하나는 준동형 암호와 준동형 메시지 인증 스킴을 결합하는 것이다. 그 중 암호화 및 인증을 동시에 진행하는 방식은 평문에 대한 암호문과 인증을 동시에 생성하여 암호문과 인증을 결합한 형태로 준동형 인증 암호의 암호문을 구성한다. 이렇게 설계한 준동형 인증암호를 HAE_{EWA}라 할 때, HAE_{EWA}는 다음과 같은 키생성(KeyGen), 암호화(Enc), 연산(Eval), 복호화(Dec) 알고리즘으로 구성된다.

HAE_{EWA}

- $KeyGen(1^{\lambda})$: 안전성 파라미터 λ에 대하여 준동형 암호의 연산키와 비밀키 $(HSE.ek, HSE.sk) \leftarrow HSE.KeyGen(1^{\lambda})$를 생성하고, 준동형 메시지 인증 스킴의 연산키와 비밀키 $(HMA.ek, HMA.sk) \leftarrow HMA.KeyGen(1^{\lambda})$를 생성한다. 최종 연산키는 $ek = (HMA.ek, HSE.ek)$로 비밀키는 $sk = (HMA.sk, HSE.sk)$로 출력한다.
- $Enc(\tau, m, sk)$: 입력 라벨 $\tau \in L$, 메시지 $m \in M$, 비밀키 $sk = (HMA.sk, HSE.sk)$에 대하여 준동형 암호의 암호문 $\tilde{c} \leftarrow HSE.Enc(m, HSE.sk)$와 준동형 메시지 인증 스킴의 인증 $\mu \leftarrow HMA.Aut(\tau, m, HMA.sk)$를 생성한다. 최종 암호문으로서 $c = (\tilde{c}, \mu)$를 출력한다.
- $Eval(f, c_1, c_2, \cdots, c_l, ek)$: 각각의 라벨과 평문 $\tau_i, m_i (1 \le i \le l)$에 대한 암호문 $c_i = (\tilde{c_i}, \mu_i)$와 함수 $f : M^l \rightarrow M$에 대하여 연산키 ek를 이용하여 $\tilde{c} = HSE.Eval(f, \tilde{c_1}, \tilde{c_2}, \cdots, \tilde{c_l}, HSE.ek)$와 $\mu = HSE.Eval(f, \mu_1, \mu_2, \cdots, \mu_l, HMA.ek)$를 계산한다. 최종 결과로 $c = (\tilde{c}, \mu)$를 출력한다.
- $Dec((f, \tau_1, \tau_2, \cdots, \tau_l), c, sk)$: 라벨 프로그램 $(f, \tau_1, \tau_2, \cdots, \tau_l)$과 암호문 $c = (\tilde{c}, \mu)$에 대하여 비밀키 $sk = (HMA.sk, HSE.sk)$를 이용 $m = HSE.Dec(\tilde{c}, HSE.sk)$과 $h = HMA.Ver((f, \tau_1, \tau_2, \cdots, \tau_l), m, \mu, HMA.sk)$를 계산한다. 만일 $h = 1$이면 m을 출력하고, $h = 0$이면 $\perp$을 출력한다.

여기서 준동형 인증 암호의 평문공간 M은 준동형 대칭키 암호와 준동형 메시지 인증 스킴의 평문공간과 동일하며, 라벨 공간 L은 준동형 메시지 인증 스킴의 라벨 공간과 동일하다.

결합 준동형 인증암호는 인증을 암호문에 단순히 결합하기 때문에 메시지에 대한 정보를 노출할 수 있다. 따라서 사용되는 준동형 암호가 구별불가능성 안전성을 만족하더라도 결합 준동형 인증암호는 다음과 같이 구별불가능성을 만족할 수 없다.

정리 1. 준동형 인증 암호 HAE_{EWA}는 선택 평문 공격자에 대한 구별불가능성(IND-CPA) 안전성을 갖지 않는다.

증명. 준동형 인증 스킴 HMA에 대하여 다음과 같이 메시지 정보를 그대로 노출하는 또 다른 준동형 인증 스킴 $\overline{HMA}$를 다음과 같이 설계할 수 있다.

$\overline{HMA}$

- KeyGen(1^{λ}) : 안전성 파라미터 λ에 대하여 준동형 메시지 인증 스킴 HMA를 이용하여 연산키와 비밀키$(ek, sk) \leftarrow HMA.KeyGen(1^{\lambda})$를 생성하고, 본 스킴의 연산키는 ek로, 비밀키는 sk로 출력한다.
- Aut(τ, m, sk) : 주어진 입력값 라벨 $\tau \in L$, 메시지 $m \in M$, 비밀키 sk에 대하여 준동형 메시지 인증 스킴 HMA를 이용하여 인증 $\mu \leftarrow HMA.Aut(\tau, m, sk)$를 계산하고 최종 인증으로서 $\overline{\mu} = (m, \mu)$를 출력한다.
- Eval$(f, \overline{\mu_1}, \overline{\mu_2}, \cdots, \overline{\mu_l}, ek)$: 함수 $f : M^l \rightarrow M$와 연산키 ek에 대하여 인증연산을 이용하여 $\mu = HMA.Eval(f, \mu_1, \mu_2, \cdots, \mu_l, ek)$를 계산한다. 그리고 $f(m_1, m_2, \cdots, m_l)$를 계산 후, 최종 결과로 $(f(m_1, m_2, \cdots, m_l), \mu)$를 출력한다.
- Ver$((f, \tau_1, \tau_2, \cdots, \tau_l), \overline{m}, \overline{\mu}, sk)$: 라벨 프로그램 $(f, \tau_1, \tau_2, \cdots, \tau_l)$과 인증 $\overline{\mu} = (m, \mu)$에 대하여 $\overline{m} = m$이면 비밀키 sk를 이용하여, $h = HMA.Ver((f, \tau_1, \tau_2, \cdots, \tau_l), m, \mu, sk)$를 출력하고, 그 이외에는 0을 출력한다.

한편, $\overline{HMA}$를 이용하여 인증과 연산단계에서 $\overline{\mu} = (m, \mu)$를 계산한 후 인증을 μ로 두고, 확인단계에서 $\overline{m} = m$를 확인하는 절차를 생략함으로

https://doi.org/10.33778/kcsa.2021.21.1.033

써 준동형 인증 스킴 HMA를 설계할 수 있다. 따라서 준동형 메시지 인증 스킴 HMA와 $\overline{HMA}$는 서로 환원될 수 있으므로 동등한 안전성을 갖는다. 결론적으로 준동형 메시지 인증 스킴 HMA를 이용하여 설계한 준동형 인증 암호와 준동형 메시지 인증 스킴 $\overline{HMA}$를 바탕으로 설계한 준동형 인증 암호는 동일한 안전성을 갖는다. $\overline{HMA}$를 바탕으로 설계한 준동형 인증 암호는 암호문이 평문을 그대로 노출하고 있어 평문 선택 공격에 대한 구별불가능성(IND-CPA)을 만족하지 않는다. 따라서 준동형 메시지 인증 스킴 HMA를 이용하여 설계한 준동형 인증 암호 HAE_{EWA}도 평문 선택 공격자에 대한 구별불가능성을 만족하지 않는다.

따름정리 1. 준동형 인증 암호 HAE_{EWA}는 선택 암호문 공격자에 대한 구별불가능성(IND-CCA)을 갖지 않는다.

한편, 위조불가능성에 대한 안전성은 그대로 계승된다. 즉 사용되는 준동형 메시지 인증 스킴이 위조불가능한 안전성을 갖는다면, 결합 준동형 인증 암호 역시 위조 불가능한 안전성을 갖는다.

정리 2. HMA가 선택 평문 공격자에 대한 위조불가능성(UF-CMA)을 안전성을 만족하면, 준동형 인증 암호 HAE_{EWA} 역시 선택 평문 공격자에 대한 위조 불가능성을 갖는다.

증명. A_1을 준동형 인증 암호 HAE_{EWA}에 대한 UF-CPA 게임의 공격자라고 할 때, 이 공격자를 이용하여 준동형 메지지 인증 스킴 HMA에 대한 UF-CMA 게임의 공격자 A_2를 설계할 수 있다. 따라서 준동형 인증 암호 HAE_{EWA}가 선택 평문 공격자에 대한 위조 불가능성(UF-CPA) 안전성을 갖지 않는다면 무시할만한 이득 이상(non-negligible advantage)으로 위조를 생성할 수 있는 공격자가 존재하고, 또한 이 공격자를 이용하여 마찬가지로 무시할만한 이득 이상으로 준동형 메시지 스킴 HMA의 위조를 생성할 수 있는 공격자가 존재한다. 즉, 준동형 메시지 스킴 HMA는 선택 메시지 공격자에 대한 위조 불가능성(UF-CMA) 안전성을 갖지 않는다. 공격자 A_1을 이용한 또 다른 공격자 A_2를 설계하는 방식은 다음과 같으며, 공격자 A_2는 공격자 A_1을 이용하기 위하여 준동형 인증 암호 HAE_{EWA}에 대한 선택 평문 공격자에 대한 위조 불가능성을 보이는 게임의 첼린처 역할을 수행한다.

우선 첼린저는 안전성 파라미터 λ에 대한 준동형 메시지 인증 스킴 HMA의 키 $HMA.ek$, $HMA.sk$를 $HMA.KeyGen(1^{\lambda})$을 이용하여 발급받고, 공격자 A_2에게 연산키 $HMA.ek$를 넘겨준다.

(Ini) 공격자 A_2는 첼린저로부터 받은 연산키 $HMA.ek$를 바탕으로 안전성 파라미터 λ에 대한 준동형 암호의 연산키와, 비밀키 $HSE.ekHSE.sk$를 $HSE.KeyGen(1^{\lambda})$를 이용하여 발급받고, 준동형 인증 암호의 연산키 $ek=(HMA.ek, HSE.ek)$를 구성하여 공격자 A_1에게 넘겨준다.

(Que) 공격자 A_1의 암호화 질의 (τ, m)에 대해서 공격자 A_2는 $\bar{c} \leftarrow HSE.Enc(m, HSE.sk)$를 계산하고, 첼린저에게 (τ, m)을 전송하여 $\mu \leftarrow HMA.Aut(\tau, m, HSE.sk)$를 수령한다. 그 후 공격자 A_1에게 $c=(\bar{c}, \mu)$로 질의에 응답한다.

(Fin) 공격자 A_2는 공격자 A_1의 위조 시도 $((f, \tau_1, \tau_2, \cdots, \tau_l), c', \mu')$를 받아 $m' \leftarrow HSE.Dec(c', HSE.sk)$을 이용, $((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu')$를 위조 시도로써 첼린저에게 제출한다. 만일 $((f, \tau_1, \tau_2, \cdots, \tau_l), c', \mu')$가 준동형 인증 암호 HAE_{EWA}에 대한 위조라면,

$Dec((f, \tau_1, \tau_2, \cdots, \tau_l), c', \mu') \neq \perp$이므로,

$HMA.Ver((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu', sk) = 1$이다. 따라서 $((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu')$는 유효한 인증임을 알 수 있다. 또한 준동형 인증 암호 HAE_{EWA}와 준동형 메시지 인증 스킴 HMA에

대해 동일한 연산 함수 f가 사용되므로 $((f, \tau_1, \tau_2, \cdots, \tau_l), c', \mu')$가 첫 번째 유형의 위조이면, $((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu')$ 역시 첫 번째 유형의 위조이다.

만일 $((f, \tau_1, \tau_2, \cdots, \tau_l), c', \mu')$가 두 번째 유형의 위조이면 각각의 τ_i와 함께 인증 및 암호화된 m_i에 대하여

$$m' = HSE.Dec(c', HSE.sk) \neq f((m_1, m_2, \cdots, m_l))$$

이므로 $((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu')$ 역시 두 번째 유형의 위조이다. 결론적으로 준동형 인증 암호에서의 위조는 준동형 메시지 스킴에서의 위조가 되므로 $Adv^{UF-CMA}_{HMA, A_2}(\lambda) \geq Adv^{UF-CPA}_{HAE_{EWA}, A_1}(\lambda)$이며, 주어진 명제의 대우를 증명하게 된다. □

따름정리 2. HMA가 선택 인증 공격자에 대한 위조 불가능성(UF-CTA)을 만족하면, 준동형 인증 암호 HAE_{EWA} 역시 선택 인증 공격자에 대한 위조 불가능성(UF-CCA)을 갖는다.

증명. 위 정리 2.에 대한 증명과정과 마찬가지로 A_1을 준동형 인증 암호 HAE_{EWA}에 대한 UF-CTA 게임의 공격자라고 할 때, 이 공격자를 이용하여 준동형 메지지 인증 스킴 HMA에 대한 UF-CCA 게임의 공격자 A_2를 설계할 수 있다. 따라서 준동형 인증 암호 HAE_{EWA}가 선택 암호문 공격자에 대한 위조 불가능성(UF-CCA)을 갖지 않는다면 무시할만한 이득 이상(non-negligible advantage)으로 위조를 생성할 수 있는 공격자가 존재하고, 또한 이 공격자를 이용하여 마찬가지로 무시할만한 이득 이상으로 준동형 메시지 스킴 HMA의 위조를 생성할 수 있는 공격자가 존재한다. 다만 위 정리에 대한 증명과정과의 차이점은 준동형 메시지 인증 스킴 HMA에 대한 선택 인증 공격자에 대한 위조 불가능성(UF-CTA) 증명을 위한 시뮬레이션을 설계를 위해 질의 단계에서 공격자의 인증에 대한 질의와 더불어 인증에 대한 확인 질의 단계를 추가로 묘사해야 된다는 점이다. 시뮬레이션(simulation) 과정은 다음과 같다.

첼린저는 λ에 대한 HMA의 키 $HMA.ek$, $HMA.sk$를 $HMA.KeyGen(1^\lambda)$을 이용하여 발급받고, 공격자 A_2에게 연산키 $HMA.ek$를 넘겨준다.

(Que) 공격자 A_1의 암호화 질의 (τ, m)에 대해서 공격자 A_2는 $\bar{c} \leftarrow HSE.Enc(m, HSE.sk)$를 계산하고, 첼린저에게 (τ, m)을 질의하여 $\mu \leftarrow HMA.Aut(\tau, m, HSE.sk)$를 수령한다. 그 후 공격자 A_1에게 $c = (\bar{c}, \mu)$로 질의에 응답한다. 공격자 A_1의 복호화 질의 $((f, \tau_1, \tau_2, \cdots, \tau_l), c', \mu)$에 대해서 공격자 A_2는 $m' \leftarrow HSE.Dec(c', HSE.sk)$를 계산하고, 첼린저에게 $((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu)$을 질의하여 그 확인 결과 h를 수령한다. 만일 $h = 1$이면 공격자 A_1에게 m'를 $h = 0$이면 $\perp$로서 질의에 응답한다.

(Fin) 공격자 A_2는 공격자 A_1의 위조 시도 $((f, \tau_1, \tau_2, \cdots, \tau_l), c', \mu')$를 받아 $m' \leftarrow HSE.Dec(c', HSE.sk)$를 계산하여, 위조 시도로서 $((f, \tau_1, \tau_2, \cdots, \tau_l), m', \mu')$를 첼린저에게 제출한다.

같은 이유로 준동형 인증 암호에서의 위조는 준동형 메시지 스킴에서의 위조가 되므로 $Adv^{UF-CTA}_{HMA, A_2}(\lambda) \geq Adv^{UF-CCA}_{HAE_{EWA}, A_1}(\lambda)$이며, 주어진 명제의 대우를 증명하게 된다.

한편, 강한 위조불가능성은 준동형 인증 암호에 계승되지 않는다. 즉 사용되는 준동형 메시지 인증 스킴이 강한 위조불가능성을 갖더라도, 결합 준동형 인증 암호는 강한 위조불가능성을 만족시킬 수 없다.

정리 3. HMA의 안전성에 관계없이, 준동형 인증 암호 HAE_{EWA}는 강한 위조불가능성(SUF-CPA, SUF-CCA) 안전성을 갖지 않는다.

증명. 준동형 인증 암호 HAE_{EWA}는 암호문 생성을 위해 암호화와 인증을 각각 생성하여 결합하므

https://doi.org/10.33778/kcsa.2021.21.1.033

로 암호화 과정과 라벨 값을 이용한 인증 생성 과정이 서로 독립적이다. 따라서 암호문 $c = (\bar{c}, \mu)$이 주어지면, 메시지 0으로부터 준동형 암호문 $c' \leftarrow HSE.Enc(0, HSE.sk)$을 생성하여 새로운 암호문 $(\bar{c} + c', \mu)$ 를 생성할 수 있다. $(\bar{c} + c', \mu)$ 는 강한 위조이므로, 준동형 인증 암호 HAE_{EWA}는 강한 위조불가능성에 대한 안전성을 만족시킬 수 없다.

4. 결 론

본 논문은 준동형 메시지 인증 스킴과 준동형 암호를 합성하여 준동형 인증 암호를 설계하였을 때 안전성 수준이 유지 또는 향상되는지 아닌지를 분석하였다. 이를 위해 관련 용어들을 정의하고 암호학의 대표적인 증명방식인 시뮬레이션(simulation)과 환원(reduction) 과정 제시함으로써 그 결과를 증명하였다. 준동형 인증 암호의 일반적인 설계 방식 중에서도 가장 효율이 좋은 결합 준동형 인증 암호에 대해 분석을 수행하였다. 결합 준동형 인증 암호는 다른 설계방식과는 달리 암호문에 대한 인증을 생성하거나, 인증에 대한 암호문을 생성하는 것이 아니라 메시지에 대한 암호문과 인증을 독립적으로 생성하여 결합하기 때문에 최종 암호문의 크기가 상대적으로 작은 장점이 있는 반면, 인증에서 메시지 정보가 드러나거나 암호문에 라벨이 적용되지 않아 쉽게 인증을 통과는 새로운 암호문을 생성할 수 있음을 알 수 있었다. 한편 사용되는 준동형 메시지 인증 스킴의 위조 불가능성은 그대로 준동형 인증 암호에도 적용됨을 확인할 수 있었다. 향후 인증화 후 암호화 방식, 암호화 후 인증 방식으로 설계되는 준동형 인증 암호에 대해서도 선택 암호문 공격자, 선택 평문 공격자를 고려한 구별 불가능성, 위조 가능성, 강한 위조 가능성에 대해 연구가 필요할 것이다.

참고문헌

[1] A. Bechrer, "Hadoop Security Design Just add Kerberos? Really?", iSEC PARTNER, 2010.

[2] O. O. Malley, K. Zhang, S. Radia, R. Marti and C. Harrel, "Hadoop Security Design", Yahoo, Incorporated, Technical Report, 2009.

[3] B. Chor, E. Kushilevitz, O. Goldreich and M. Sudan, "Private Information Retrieval", Journal of the ACM, Vol. 45, No. 6, pp. 956-981, 1998.

[4] D. Boneh, E. Kushilevitz, R. Ostrovsk and W. E. Skeith, "Public Key Encryption that Allows PIR Queries", CRYPTO 2007, Vol. 4622, pp. 50-67, 2007.

[5] H. Avni, S. Dolev, N. Gilboa and X. Li, "SSSDB: Database with Private Information Search", International Workshop on Algorithmic Aspects of Cloud Computing, Vol. 9511, pp.49-61, 2015.

[6] D. Song, D. Wagner, and A. Perrig, "Practical Techniques for Searches on Encrypted Data", Proceeding of IEEE Symposium on Security and Privacy, pp. 44-55, 2000.

[7] J. Benaloh, M. Chase, E. Horvitz, and K.Lauter, "Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records", Proceeding of the ACM Workshop on Cloud Computing Security, pp.103-114, 2009.

[8] Q. Liu, G. Wang and J. Wu, "Secure and Privacy Preserving Keyword Searching for Cloud Storage Services", Journal of network and computer applications, Vol. 35, pp. 927-933, 2012.

[9] K. Pasupuleti, S. Ramalingam, and R. Buyya, "An Efficient and Secure Privacy-preserving Approach for Outsourced Data of Resource Constrained Mobile Devices in Cloud Computing", Journal of network and computer applications, vol. 64, pp. 12-22, 2016.

[10] S. Gajek, "Dynamic Symmetric Searchable Encryption from Constrained Functional Encryption", Proceeding of CT-RSA 2016, Vol. 9610, pp. 75-89, 2016.

[11] M. Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "Fully homomorphic encryption over the integers", Proceeding of EUROCRYPT 2010, Vol. 6110, pp.

24-43, 2010.

[12] Z. Brakerski, "Fully homomorphic encryption without modulus switching from classical GapSVP", Proceeding of CRYPTO 2012, Vol. 7417, pp. 868-886, 2012.

[13] Z. Brakerski, C. Gentry, and S. Halevi, "Packed ciphertexts in LWE-based homomorphic encryption", Proceeding of PKC 2013, Vol. 7778, pp. 1-13. 2013.

[14] Z. Brakerski and V. Vaikuntanathan, "Fully homomorphic encryption from Ring-LWE and security for key dependent messages", Proceeding of CRYPTO 2011, Vol. 6841, pp. 505-524, 2011.

[15] T. Krovetz and P. Rogaway, "The software performance of authenticated-encryption modes", Proceeding of Fast Software Encryption, Vol. 6733, pp. 306-327, 2011.

[16] P. Rogaway, M. Bellare, and J. Black, "OCB: A block-cipher mode of operation for efficient authenticated encryption", ACM Transaction on Information and System Security, pp. 365-403, 2003.

[17] C. Joo, A. Yun, "Homomorphic authenticated encryption secure against chosen-ciphertext attack", International Conference on the Theory and Application of Cryptology and Information Security. pp. 173-192, 2014.

[18] J. Cheon, K. Han, S. Hong, H. Kim, J. Kim, Y. Song, "Toward a secure drone system: Flying with real-time homomorphic authenticated encryption", IEEE access, Vol. 6, pp.24325-24339, 2018.

[19] P. Struck, L. Schabhüser, D. Demirel, J. Buchmann, "Linearly homomorphic authenticated encryption with provable correctness and public verifiability", International Conference on Codes, Cryptology and Information Security, pp.142-160, 2017.

[20] M. Bellare and C. Namprempre, "Authenticated encryption: Relations among notions and analysis of the generic composition paradigm", Journal of Cryptology, pp. 469-491, 2008.

[저 자 소 개]

김 진 수 (Jinsu Kim)
해군사관학교 공학사
서울대학교 수리과학부 이학석사
서울대학교 수리과학부 이학박사
현) 해군사관학교 수학과 부교수
email : nemokjs1@gmail.com