특집논문-08-13-1-06

전방향 안전성을 보장하는 공개키 브로드캐스트 암호 기법†

박 종 환[a)‡], 윤 석 구[a)]

Forward-Secure Public Key Broadcast Encryption

Jong Hwan Park[a)‡], and Seok Koo Yoon[a)]

요 약

본 논문에서는 전방향 안전성(forward-secrecy)을 보장하는 공개키 브로드캐스트 암호 기법을 제안한다. 공개키 브로드캐스트 암호는 공개키를 이용하여 구성원 누구나 메시지를 전송할 수 있고, 탈퇴자 그룹을 효율적으로 배제(revocation)할 수 있는 기법이다. 여기에 전방향 안전성을 보장하려는데, 전방향 안전성은 사용자의 비밀키가 노출되더라도 그 노출된 시점 이전의 암호문을 쉽게 복호화할 수 없도록 하는 것이다. 이러한 기능이 없다면 권한 없는 수신자가 과거의 방송을 수집하고 이후 정당한 비밀키를 받아서 과거의 방송을 복호화할 수 있는 문제가 발생한다. 전방향 안전성은 특히 유료 방송 등의 환경에서 요구된다. 본 논문에서 제안되는 기법은 2005년 Boneh-Boyen-Goh가 제시한 계층구조의 신원 기반 암호기법을 변형하여 설계된다. 먼저 BBG기법을 사용하여 새로운 공개키 브로드캐스트 암호기법을 설계하고, 다시 BBG기법에서 사용된 하위레벨 비밀키 생성 알고리즘을 사용하여 전방향 안전성을 부여한다. 제안되는 기법은 타원곡선 위의 페어링(pairing)을 이용하여 설계되며, 전체 사용자 n에 대하여 $O(\sqrt{n})$ 사이즈의 통신량과 비밀키 저장량을 가진다. 특히 비밀키 저장량은 탈퇴자 수가 증가할수록 줄어드는 장점을 가진다. 통신량이 중요한 환경에서는 이전에 제시된 기법보다 본 논문에서 제안된 기법을 사용하는 것이 더 바람직한데, 이는 통신량은 동일하지만 비밀키 저장량이 더 적기 때문이다. 제안된 기법은 Bilinear Diffie-Hellman Exponent 가정 하에서 선택 암호문 공격에 안전하도록 설계되며, 그 증명은 랜덤 오라클을 사용하지 않는다.

ABSTRACT

Public Key Broadcast Encryption (PKBE) allows a sender to distribute a message to a changing set of users over an insecure channel. PKBE schemes should be able to dynamically exclude (i.e., revoke) a certain subset of users from decrypting a ciphertext, so that only remaining users can decrypt the ciphertext. Another important requirement is for the scheme to be forward-secrecy. A forward-secure PKBE (fs-PKBE) enables each user to update his private key periodically. This updated private key prevents an adversary from obtain the private key for certain past period, which property is particularly needed for pay-TV systems. In this paper, we present a fs-PKBE scheme where both ciphertexts and private keys are of $O(\sqrt{n})$ size. Our PKBE construction is based on Boneh-Boyen-Goh's hierarchical identity-based encryption scheme. To provide the forward-secrecy with our PKBE scheme, we again use the delegation mechanism for lower level identities, introduced in the BBG scheme. We prove chosen ciphertext security of the proposed scheme under the Bilinear Diffie-Hellman Exponent assumption without random oracles.

Keywords : Public Key Broadcast Encryption, Forward-Secrecy.

a) 고려대학교 정보경영공학전문대학원
Graduate School of Information Management and Security, Korea University
‡ 교신저자 : 박종환(decartian@cist.korea.ac.kr)
※본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (IITA-2008-(C1090-0801-0025))

Ⅰ. 서 론

최근 브로드캐스트(broadcast) 암호를 위성 TV나 저작권 보호를 위한 알고리즘에 적용하는 연구가 진행되고 있다.

브로드캐스트 암호는 수신 권한이 있는 정당한 수신자 만이 브로드캐스트 메시지를 복호화할 수 있고, 수신 권한이 없는 제삼자는 메시지에 대한 정보를 얻을 수 없도록 한다. 브로드캐스트 암호는 크게 대칭키 기반과 공개키 기반으로 구분되는데, 대칭키 기반은 지정된 자만이 메시지를 전송할 수 있고 공개키 기반은 공개키에 접근 가능한 자는 누구나 메시지를 전송할 수 있다. 전자는 빠른 연산이 장점이나 가입과 탈퇴를 처리하는 데 복잡한 비밀키 갱신과정이 필요하다는 단점이 있고, 후자는 가입과 탈퇴의 처리는 효율적이나 복호화에 필요한 계산이 느리다는 단점이 있다. 그러나 최근 타원곡선 위의 페어링(Pairing)을 효율적으로 구현하는 알고리즘이 제시되면서, 후자의 단점을 보완할 수 있는 페어링 기반의 공개키 브로드캐스 암호기법이 주목받고 있다.

일반적으로 공개키 브로드캐스트 암호기법은 하이브리드(Hybrid) 암호 메커니즘을 제공하는데 사용된다. 하이브리드 암호 메커니즘이란 전송하고자 하는 메시지를 대칭키(Symmetric) 기법으로 암호화한 후 그 대칭키를 공개키 브로드캐스트 암호(Public Key Broadcast Encryption. 이하에서는 PKBE로 표기한다) 기법을 사용하여 암호화하는 것을 말한다. 이때 대칭키를 암호화한 부분을 헤더(Header)라 할 때, 실제 송신자가 전송하는 총 메시지는 (S, Header, CM)이 된다. 여기서 S는 메시지를 수신하기에 정당한 권한이 있는 수신자의 집합을 의미하고, CM은 Header로부터 도출되는 대칭키로 실제 메시지 M을 암호화한 것이다. S에 속하는 정당한 수신자는 헤더로부터 대칭키를 얻고 그 키를 이용하여 CM을 복호화하는 과정을 거친다.

PKBE기법이 제공하는 핵심기능은 유동적으로 변하는 가입과 탈퇴의 문제를 효율적으로 처리하는 것이다. 즉 앞서 언급한 총 사용자 중에서 정당한 수신자의 집합 S가 변하는 경우인데, 특히 탈퇴의 경우가 문제가 된다. 이는 탈퇴자 한 명이 정당한 메시지를 수신할 수 없어야 함은 물론이고, 탈퇴자들이 공모(Collusion)하는 경우에도 정당한 메시지를 수신할 수 없도록 해야 한다는 것을 의미한다. 바로 탈퇴자들의 공모(Collusion) 공격에도 안전하도록 설계되어야 한다는 것이 PKBE기법의 안전성을 정의하는데 중요한 사항이 된다. 처음 제시되었던 PKBE기법[1]은 미리 설정된 임계값(Threshold) t에 대하여 t명 이하까지 탈퇴자를 효율적을 처리하는 기법이었지만, 최근 들어 Boneh, Gentry, Waters[2]가 페어링(Pairing)을 이용하여 탈퇴자 수에 제한 없이 공모 공격에 안전한(Fully collusion-secure) PKBE기법을 제시하였다.

PKBE기법의 안전성에서 고려해야 될 또 다른 사항으로 전방향 안전성(Forward secrecy)이 있다. 이는 간단히 말해 '과거의 통신을 보호한다'는 것으로서, 일정 시점에 사용자의 비밀키가 노출되더라도 그 노출시점 이전의 통신으로부터 공격자는 필요한 정보를 얻을 수 없도록 한다는 것이다. 이러한 성질은 특히 방송에서 필요하다. 예를 들어, 7월까지 정당한 수신 권한이 없는 자가 방송을 수집하고 저장한 뒤 8월에 정당한 수신자로 가입하고 비밀키를 받는다고 하자. 부여된 비밀키에 전방향 안전성을 보장하는 장치가 설치되지 않았다면 그 비밀키를 이용하여 7월까지의 저장된 방송을 쉽게 복호화할 수 있게 된다. 이는 비밀키가 분실된 경우에도 그대로 적용될 수 있는 시나리오이다.

이러한 전방향 안전성을 해결하기 위해 Yao 등[3]이 PKBE기법에 계층구조의 신원기반 암호(Hierarchical Identity-Based Encryption. 이하에서는 HIBE로 표기한다) 기법을 적용하여 전방향 안전성을 보장할 수 있음을 보였다. 그들의 아이디어는 신원기반의 ID정보를 시간에 대한 정보로 대체하고, HIBE기법의 하위 레벨 ID에 대한 비밀키 생성 알고리즘을 이용하여 시간의 진화에 대응하는 것이다. 이러한 접근은 최근 Attrapadung 등[4]이 BGW기법에 기존에 제시된 HIBE기법을 적용하여 전방향 안전성을 가지는 PKBE(Forward-Secure PKBE. 이하에서는 fs-PKBE라 표기한다) 기법을 제시하면서 구체화되었는데, 그들의 기법은 BGW기법의 영향을 받아서 총 n명의 사용자에 대해 $O(\sqrt{n})$의 비밀키와 통신량 사이즈를 갖는 것이 특징이다.

본 연구에서는 새로운 fs-PKBE기법을 제시한다. 제안되는 기법은 페어링 기반 위에서 설계되며, Boneh, Boyen, Goh[5]가 제안한 HIBE기법을 응용하여 얻어진다. 새롭게 제안되는 기법의 효율성은 Attrapadung 등[4]이 제안한 기법과 유사하다. 그러나 통신량 사이즈가 중요한, 즉 통신량을 줄이는 것이 유리한, 응용환경에서는 본 연구에서 제안되는 기법이 더 바람직한데 이는 저장량에서 더 좋은 결과를

갖기 때문이다. 특히 탈퇴자가 증가할수록 개별 사용자의 비밀키 사이즈가 줄어드는 장점을 가진다. 구체적인 성능 비교는 제IV절에서 제시될 것이다. 제안되는 기법은 전방향 안전성과 선택 암호문 공격에 안전한 것으로 증명되는데, 선택 암호문 공격에 안전하도록 설계하기 위해 Boyen 등[6]이 제시한 해쉬함수 기반 변환방법의 아이디어를 사용한다. 안전성 증명은 랜덤 오라클(random oracle)을 사용하지 않는 환경에서 이루어진다.

II. 제안된 기법의 이해를 위한 사전지식

1. HICBE와 그 안전성 모델

Attrapadung 등[4]이 제안한 fs-PKBE기법은 우선 계층구조의 신원 정보를 가미한 PKBE(Hierarchical Identity-Coupling Broadcast Encryption. 이하에서는 HICBE라 표기한다) 기법으로부터 유도된다. 계층구조에서의 ID정보를 시간에 대한 정보로 바꾼 후 간단한 변환방법을 사용하면 전방향 안전성을 얻을 수 있다. 구체적인 변환 방법은 [4]을 참조하기 바란다. 본 연구에서는 HICBE기법을 제안한다. HICBE기법에서 사용자는 (i, ID)로 특정된다. 여기서 n명의 총 사용자 수에 대해 $i \in \{1,\ldots,n\}$이고, 미리 설정된 자연수 L에 대하여 $ID=(I_1,\ldots,I_k),\ k=1,\ldots,L$이다. 부연하면, L은 계층구조에서 ID 벡터의 최대 길이를 나타낸다. 수식표기의 편의를 위해 $ID_{|j}$는 $(I_1,\ldots,I_j)$을 나타낸다고 하자. 먼저 HICBE기법을 구성하는 알고리즘은 다음과 같다.

- *Setup*(n, L): 총 사용자 수 n과 ID 정보가 가질 수 있는 최대 길이 L을 입력한다. 이에 대해 공개키 PK와 마스터 키 mk를 출력한다.
- Pr*ivKeyGen*(PK,mk,i,ID): 공개키 PK, 마스터 키 mk, 사용자 번호 i, 그리고 ID를 입력한다. 이에 대해 비밀키 d_i 또는 $d_{i,ID}$를 출력한다.
- *Derive*$(PK,i,ID,d_{ID_{|k-1}})$: 공개키 PK, 사용자 번호 i, ID, 그리고 $ID_{|k-1}$의 비밀키 $d_{i,ID_{|k-1}}$를 입력한다. 이에 대해 $d_{i,ID}$를 출력한다.
- *Encrypt*(PK,S,ID): 공개키 PK, 수신자 집합 $S\subseteq\{1,\ldots,n\}$, ID를 입력한다. 이에 대해 한 쌍의 (Hdr,K)를 출력한다. 여기서 Hdr는 헤더이고, K는 메시지 암호화 키(앞에서 언급한 대칭키)이다.
- *Decrypt*$(d_{i,ID},S,Hdr,PK)$: 사용자 번호 i와 ID에 대한 비밀키 $d_{i,ID}$, 수신자 집합 S, 헤더 Hdr, 그리고 공개키 PK를 입력한다. 이에 대해 $i\in S$이면, 메시지 암호화 키 K를 출력한다.

다음으로 HICBE기법의 안전성을 정의한다. 선택 암호문 공격에 대한 안전성은 공격자 A와 챌린저(challenger) C사이에 약속된 다음과 같은 게임으로 설명된다. 여기서 A와 C는 총 사용자 수 n과 ID벡터의 최대 길이 L을 사전에 알고 있다고 가정하자.

- *Init*: A는 자신이 공격하고자 하는 수신자 집합 $S^*\subseteq\{1,\ldots,n\}$과 ID^*를 결정한다.
- *Setup*: C는 *Setup* 알고리즘을 돌려서 공개키 PK와 마스터 키 mk를 얻는다. A에게 PK를 준다.
- *Query Phase* 1: A는 전략적으로 다음의 질의들을 던진다.
  - (i,ID)에 대한 비밀키 질의. 여기서 $i\notin S^*$이거나, $i\in S^*$이면 $ID\neq ID^*$ 또는 ID는 ID^*의 선행벡터(prefix)가 아니어야 한다. 부연하면, $i\notin S^*$인 경우에는 $ID=ID^*$이거나 ID^*의 선행벡터이어도 된다. C는 Pr*ivKeyGen* 또는 *Derive* 알고

리즘을 돌려 (i,ID)에 대한 비밀키 $d_{i,ID}$를 얻고, 이를 A에게 준다.

- (i,ID,S,Hdr)로 구성된 복호화 질의. 여기서 $i \in S \subseteq S^*$이다. C는 *Decrypt* 알고리즘을 돌려서 나온 결과값을 A에게 준다.

● *Challenge*: C는 $Encrypt(PK,S^*,ID^*)$를 돌려서 (Hdr^*,K)를 얻는다. 그 다음 C는 무작위로 $b \in \{0,1\}$을 선택한다. 만일 $b=1$이면 $K^*=K$로 놓고, $b=0$이면 K와 길이가 같은 난수열을 K^*로 놓는다. 마지막으로 C는 도전문제로서 (Hdr^*,K^*)를 A에게 준다.

● *Query Phase* 2: A는 다음의 질의들을 던진다.

- (i,ID)에 대한 비밀키 질의. C는 *Query Phase* 1에서의 같이 응답한다.
- (i,ID,S,Hdr)로 구성된 복호화 질의. 여기서 $Hdr \neq Hdr^*$이어야 한다. 마찬가지로 C는 *Query Phase* 1에서의 같이 응답한다.

● *Guess*: A는 그의 추측한 값 $b' \in \{0,1\}$을 출력한다. 만일 $b=b'$이면 A는 이긴다.

위 게임에서의 공격자 A를 $IND-sID-sSet-CCA$ 공격자로 명명한다. 그리고 HICBE기법 E를 공격하는 A의 이점(advantage)을 $Adv_{E,A} \equiv |\Pr[b=b'] - 1/2|$로 정의한다.

정의 1. HICBE기법에 대하여 공격시간 t를 갖고, 최대 q_P개의 비밀키 질의와 q_D개의 복호화 질의를 던지는 공격자 A가 $Adv_{E,A} < \varepsilon$를 만족하면, HICBE기법은 $(t,\varepsilon,n,q_p,q_D)-IND-sID-sSet-CCA$에 안전하다고 한다.

2. 페어링(Pairing)과 계산 복잡도 가정

기본적으로 페어링에 대해서는 [7,5]의 표현을 따른다. G와 G_T를 곱셈연산을 가지고 소수 위수를 가지는 순환그룹이라 하자. g를 G의 생성원이라 하자. 페어링 $e: G \times G \rightarrow G_T$는 다음의 조건을 만족하는 함수이다.

1) Bilinear: 모든 원소 $u,v \in G$와 $a,b \in Z$에 대하여 $e(u^a,v^b)=e(u,v)^{ab}$를 만족한다.
2) Non-degenerate: e는 $e(g,g) \neq 1$을 만족한다.
3) Computable: e를 효율적으로 계산하는 알고리즘이 존재한다.

다음으로 결정적인 (decisional) $(b+1)$-Bilinear Diffie-Hellman Exponent(BDHE)을 설명한다. 이 가정은 이미 [5,2] 등에서 안전성을 증명하기 위해 사용되었다. 결정적인 $(b+1)$-BDHE문제는 다음과 같이 정의된다. 먼저 $g_i = g^{(\alpha^i)}$라 하고, $\vec{g}_{\alpha,b} = (g_1,\ldots,g_b,g_{b+2},\ldots,g_{2b})$라 하자. 이때 $(z,g,\vec{g}_{\alpha,b},T)$를 입력 받았을 때 공격자는 $T=e(z,g_{b+1})$인지 랜덤한 수인지를 결정하는 문제이다. 그룹 G에서 결정적인 $(b+1)$-BDHE문제를 푸는 공격자 A의 이점(advantage)은 아래와 같을 때, A는 ε의 이점을 가진다고 한다.

$$|\Pr[A(z,g,\vec{g}_{\alpha,b},e(z,g_{b+1}))=0] - \Pr[A(z,g,\vec{g}_{\alpha,b},T)=0]| \geq \varepsilon.$$

여기서 확률은 Z_p에서 α를, G_T에서 T를, G에서 z를 랜덤하게 선택하는 것과 A의 랜덤한 테이프 유지로 계산된다.

정의 2. 그룹 G에서 결정적인 $(b+1)$-BDHE문제를 푸는데 공격시간 t를 갖는 어떠한 공격자 A라도 그 이점이 적어도 ε보다 크지 않으면, $(t,\varepsilon,b+1)$-BDHE

가정은 그룹 G 에서 유효하다고 한다.

Ⅲ. 새로운 HICBE 기법의 설계

본 장에서는 새로운 HICBE 기법을 제안한다. 그 기법은 전방향 안전성과 선택 암호문 공격에 안전하도록 설계된다. 특히 선택 암호문 공격에 안전하도록 설계하기 위해 Boyen, Mei, Waters[6]가 제안한 해쉬함수 기반의 아이디어를 이용한다. 지금까지 (랜덤 오라클을 사용하지 않는 증명에서) 선택 암호문 공격에 안전하도록 설계하는 방법은 1) 일회용 서명 기반[8], 2)메시지 인증 코드(Message Authentication Code: MAC) 기반[9], 3)해쉬함수 기반[6]의 세 가지가 있다. 이 가운데서 해쉬함수를 이용하는 방법은 암호문에 서명이나 MAC을 첨부하지 않으므로 암호문의 길이가 크게 팽창되지 않는다는 장점이 있다. 이러한 BMW 기법을 사용하기 위해 충돌 저항성을 가지는 해쉬함수 $H_k : G \to Z_p$ 가 필요하다. 여기서 $k \in K$ 로 키 공간 K 에서 임의로 선택된 키이다. 해쉬함수의 안전성에 대해 다음과 같이 정의한다: $H_k(x) = H_k(y)$ 을 만족하는 충돌쌍 x, y 를 찾기 위해 시간 t 를 가지는 공격자가 확률 ε 이하로 성공할 수 있을 때, 해쉬함수 집합은 (t, ε)-충돌 저항성을 가진다고 한다.

총 사용자 $n(=ab)$ 명에 대해 [2]에 나온 아이디어를 적용한다. 즉 총 사용자 n 을 a 개의 부분집합으로 분할하고, 각각의 집합은 b 명의 사용자를 수용할 수 있다. 이를 간단히 b-HICBE기법으로 표기한다. 또한 ID 벡터의 최대길이 L 에 대하여 $L \le b$ 라고 가정하자.

1. 알고리즘 설명

● *Setup*(n, L): b-HICBE기법의 파라미터를 생성하기 위해, 먼저 생성원 $g \in G$ 를 선택한다. 그리고 $\alpha \in Z_p$ 를 랜덤하게 선택하고, $g_1 = g^{\alpha}$ 로 놓는다. 다음으로 랜덤하게 $h, h_1, x_0, \ldots, x_a,\ y_1, \ldots y_b \in G$ 을 선택한다. 또한 해쉬함수 H 를 위한 랜덤한 해쉬키 $k \in K$ 을 선택한다. 공개키 PK 와 마스터 키 mk 는 다음과 같이 주어진다.

$$PK = (g, g_1, h, h_1, x_0, \ldots, x_a, y_1, \ldots y_b) \in G^{a+b+5},$$
$$mk = h^{\alpha}.$$

여기서 PK 에는 (G, G_T, e, p, H) 에 대한 정보가 포함된다.

● Pr*ivKeyGen*(PK, mk, i, ID): $i \in \{1, \ldots n\}$ 에 대한 루트 비밀키를 계산하기 위해, 먼저 $i = (u-1)b + v$ 를 만족하는 $u, v (1 \le u \le a$ 이고 $1 \le v \le b)$를 구한다. 다음 난수 $r \in Z_p$ 을 뽑고, i 에 대한 비밀키를 다음과 같이 계산한다.

$$d_i = (h^{\alpha}(x_u y_v)^r, h_1^r, g^r,\ y_1^r, \ldots, y_{v-1}^r, y_{v+1}^r, \ldots, y_b^r) \in G^{b+3}.$$

$ID = (I_1, \ldots I_k) \in Z_p^{\ k}, k \le L$ 인 (i, ID) 에 대한 비밀키를 생성하기 위해서는 두 개의 난수 $r, r' \in Z_p$ 을 뽑고 비밀키를 아래와 같이 계산한다.

$$d_{i,ID} = (h^{\alpha}(x_u y_v)^r (x_0 y_1^{I_1} \cdots y_k^{I_k})^{r'}, h_1^r, g^r, y_1^r, \ldots,$$
$$y_{v-1}^r, y_{v+1}^r, \ldots, y_b^r, g^{r'}, y_{k+1}^{r'}, \ldots, y_l^{r'})$$

● *Derive*$(PK, i, ID, d_{i,ID_{|k-1}})$: 이 알고리즘은 [5]에 제시된, 하위 레벨을 위한 키 생성 알고리즘과 같다. 먼저 $ID_{|k-1} = (I_1, \ldots, I_{k-1})$ 에 대한 비밀키가 아래처럼 주어진다고 하자.

$$d_{i,ID_{|k-1}} = (h^{\alpha}(x_u y_v)^r (x_0 y_1^{I_1} \cdots y_{k-1}^{I_{k-1}})^{r''}, h_1^r, g^r, \ldots, g^{r''},$$
$$y_k^{r''}, \ldots, y_l^{r''})$$
$$= (d_1, d_2, d_3, \ldots, \pi_0, \pi_k, \ldots, \pi_l)$$

$d_{i,ID}$ 을 위해 난수 $r^* \in Z_p$ 을 선택하고, 아래와 같이 출력한다.

$$d_{i,ID}=(d_1\pi_k^{I_k}\cdot(x_0y_1^{I_1}\cdots y_k^{I_k})^{r^*}, d_2,\ldots,\pi_0 g^{r^*},\pi_{k+1}y_{k+1}^{r^*},\ldots,\pi_l y_l^{r^*})$$

여기서 $r'=r''+r^*$임을 알 수 있고, 난수 r^*에 의해 새롭게 만들어진 비밀키 $d_{i,ID}$는 정당하게 분포된 키임을 알 수 있다.

- *Encrypt*(PK,S,ID) : 송신자는 난수 $s \in Z_p$를 뽑고, 메시지 암호화 키 $K=e(h,g_1)^s \in G_T$로 놓는다. 다음 g^s를 계산하고 $\mu=H_k(g^s)\in Z_p$를 계산한다. $ID=(I_1,\ldots I_k)$라 하자. 이 때 헤더는 아래와 같이 계산된다.

$$Hdr=(\,(x_1h_1^{\mu}\prod_{j\in S_1}y_j)^s,\ldots,(x_ah_1^{\mu}\prod_{j\in S_a}y_j)^s,\; g^s,\;(x_0y_1^{I_1}\cdots y_k^{I_k})^s\,)\in G^{a+2}$$

이 알고리즘은 (Hdr,K)를 출력한다.

- *Decrypt*$(d_{i,ID},S,Hdr,PK)$: i가 집합 S_u에서 v의 인덱스를 받는다고 하자. i가 자신의 비밀키 $d_{i,ID}=(d_1,d_2,d_3,k_{i,1},\ldots k_{i,v-1},\; k_{i,v+1},\ldots,k_{i,b},\pi_0,\pi_{k+1},\ldots,\pi_l)$ 자를 가지고 복호화한다고 하자. $Hdr=(A_1,\ldots,A_a,B,C)$라고 하고, $ID=(I_1,\ldots I_k)$라 하자. 먼저, $\mu'=H_k(B)$를 계산하고 아래의 두 페어링 등식이 성립하는지 체크한다.

$$e(A_u,g)=e(x_uh_1^{\mu'}\prod_{j\in S_u}y_j,B),\qquad e(C,\,g)=e(x_0y_1^{I_1}\cdots y_k^{I_k},B)$$

만일 하나라도 성립하지 않으면, $\perp$를 출력한다. 두 식이 모두 성립하면, 다음을 출력한다.

$$K=e(d_1d_2^{\mu'}\prod_{\substack{j\in S_u\\ j\neq v}}k_{i,j},B)\,/\,e(A_u,d_3)e(C,\pi_0).$$

위에서 제안된 *b-HICBE*기법의 정확성을 살펴보기로 하자. 먼저 *Decrypt* 알고리즘에 필요한 원소들 $(d_1d_2^{\mu}\prod_{j\in S_u,j\neq v}k_{i,j},d_3,\pi_0)$는 다음과 같이 분포된다.

$$(h^{\alpha}(x_uh_1^{\mu'}\prod_{j\in S_u}y_j)^{r_1}\cdot(x_0y_1^{I_1}\cdots y_k^{I_k})^{r_2},\,g^{r_1},\,g^{r_2})$$

여기서 r_1,r_2은 Z_p에서 랜덤하게 분포된다. 다음으로 메시지 암호화 키 K가 바르게 계산되는지 여부는 아래의 식에서 알 수 있다.

$$\begin{aligned}K&=\frac{e(d_1d_2^{\mu'}\prod_{j\in S_u,j\neq v}k_{i,j},B)}{e(A_u,d_3)e(C,\pi_0)}\\&=\frac{e(h^{\alpha}(x_uh_1^{\mu'}\prod_{j\in S_u}y_j)^{r_1}\cdot(x_0y_1^{I_1}\cdots y_k^{I_k})^{r_2},\,g^s)}{e((x_uh_1^{\mu'}\prod_{j\in S_u}y_j)^s,\,g^{r_1})\,e((x_0y_1^{I_1}\cdots y_k^{I_k})^s,\,g^{r_2})}\\&=e(h,g_1)^s\end{aligned}$$

다음으로 *Decrypt* 알고리즘에서 암호문의 정확성을 체크하고 복호화하기 위해서는 총 일곱 번의 페어링이 필요하다. 그러나 [10]에서 사용된 아이디어를 적용하면 실제 페어링은 세 번으로 줄어든다. 이를 위해서는 우선 r_1, $r_2 \in Z_p$을 랜덤하게 뽑고, 다음을 계산한다.

$$\tilde{d}_{1,2}=d_1d_2^{\mu'}\prod_{\substack{j\in S_u\\ j\neq v}}k_{i,j}\cdot(x_uh_1^{\mu'}\prod_{j\in S_u}y_j)^{r_1}\cdot(x_0y_1^{I_1}\cdots y_k^{I_k})^{r_2},$$

$$\tilde{d}_3=d_3\cdot g^{r_1},\qquad \tilde{\pi}_0=\pi_0\cdot g^{r_2}.$$

그리고 $e(\tilde{d}_{1,2},B)/e(A_u,\tilde{d}_3)e(C,\tilde{\pi}_0)$을 출력한다.

2. 제안된 기법의 안전성

사용자 제안된 b-HICBE기법이 선택 암호문 공격에 안전하다는 것은 $(b+2)$-BDHE가정과 일회용 서명의 위

조 어려움을 기반으로 보일 수 있다. 구체적인 증명은 생략한다.

정리 1. $(t_1,\varepsilon_1,b+2)$-BDHE 가정이 그룹 G에서 유효하고, 해쉬함수의 집합 $\{H_k\}$이 (t_2,ε_2)-충돌 저항성을 가진다고 가정하자. 그러면 위에서 제안된 b-HICBE기법은 $(t_3,\varepsilon_3,n,q_p,q_D)-IND-sID-sSet-CCA$에 안전하다. 시간과 이점에 대한 관계는 아래와 같다.

$$t_3 < t_1 - O(\tau bLq_p), \quad \varepsilon_1+\varepsilon_2 \geq \varepsilon_3.$$

여기서 τ는 G에서 한 번의 지수승(exponentiation)을 하는데 필요한 최대 시간이다.

IV. 제안된 HICBE 기법의 효율성

n을 총 사용자 수라 하고, n은 a개의 부분집합으로 나뉘어지고 각각의 집합은 b명을 포함한다. R은 탈퇴자의 집합이라 하고 $r=|R|$은 탈퇴자의 수라 하자. [5]과 같이 탈퇴자의 수가 적다면, 즉 $r<<n$라면, 송신자는 원래의 (S,ID,Hdr,C_M) 대신에 (R,ID,Hdr,C_M)을 전송할 수 있다. 여기서 Hdr는 여전히 정당한 수신자 집합 S에 대하여 구성된다. 이 경우 (i,ID)가 갖는 비밀키는 아래와 같이 구성된다.

$$d_{i,ID} = (h^{\alpha}(x_u y_v)^{r_1}(x_0 y_1^{I_1}\cdots y_k^{I_k})^{r_2}(\prod_{j\neq v}^{b} y_j)^{r_1},\ h_1^{r_1},$$
$$g^{r_1}, y_1^{r_1},\ldots,y_{v-1}^{r_1}, y_{v+1}^{r_1},\ldots,y_b^{r_1}, g^{r_2}, y_{k+1}^{r_2},\ldots)$$

탈퇴자 집합 R이 $R_1,\ldots,R_a$의 부분집합들로 나뉘어진다고 하자. $i(=(u-1)b+v)$가 속한 S_u내의 R_u에 대하여, $i\notin R_u$라고 하자. 이 경우 i는 비밀키 원소 중 $(y_1^{r_1},\ldots,y_{v-1}^{r_1},y_{v+1}^{r_1},\ldots,y_b^{r_1})$를 이용하여 R_u에 대응되는 원소들의 곱 $\theta=\prod_{j\in R_u} y_j^{r}$을 계산한다. 이어서 비밀키의 첫 번째 성분으로부터 θ값을 나누면, 첫 번째 성분은

$$h^{\alpha}(x_u\prod_{\substack{j=1\\ j\notin R_u}}^{b} y_j)^{r_1}(x_0 y_1^{I_1}\cdots y_k^{I_k})^{r_2}$$

이 된다. 그 후 $\{y_j^{r_1}\}_{j\in R_u}$의 원소들은 비밀키에서 제거될 수 있다. 새로운 탈퇴자 집합 R'(그러므로 R'_u)에 대해서는 R'_u와 남아있는 원소들 $\{y_j^{r_1}\}_{j\in S_u\setminus R_u}$과 비교하여, $\{y_j^{r_1}\}_{j\in(S_u\setminus R_u)\cap R'_u}$의 원소들을 확인하고 이를 또 제거한다. 이 과정은 탈퇴자가 새롭게 증가할수록 반복된다. 그러므로, R(그러므로 R_u)의 크기가 커질수록 비밀키의 사이즈는 작아진다.

[4]에서 제안된 HICBE기법은 Boneh, Gentry, Waters가 제안한 PKBE을 기반으로 하여 설계되었다. 이를 $HICBE_{BGW}$라 표기한다. 원래 BGW-PKBE기법은 총 사용자 $n(=ab)$을 b명이 수용되는 a개의 집합으로 분할할 수 있으므로, [4]에서 제안된 HICBE기법도 a개의 부분 집합을 갖도록 확장할 수 있다. 이를 $b-HICBE_{BGW}$로 나타낸다. $b-HICBE_{BGW}$과 본 논문의 제안된 기법과의 가장 큰 차이점은 PK의 개수와 복호화 알고리즘의 입력값에 있다. 먼저 PK 사이즈를 보면, $b-HICBE_{BGW}$는 $(a+2b+2)$개의 그룹 G의 원소들로 이루어진다. 그러나 본 논문에서 제안된 기법은 $(a+b+5)$개의 그룹 원소로 이루어지는데, 이 차이는 비밀키를 구성하는 대수적인 구조의 차이에 기인한다. 또한 복호화 알고리즘의 입력값을 보면, $b-HICBE_{BGW}$는 PK를 입력하는 것이 요구되나. 제안된 기법은 PK를 요구하지 않는다. 여기서 PK는 공개된 정보이므로 전송 시에 헤더에 포함되거나, 사용자의 저장매체에 추가적으로 저장될 수 있다. 따라서 $b-HICBE_{BGW}$은 (복호화에 필요한) PK가 어디에 포함되는지에 따라 두 가지로 구분되는데, 1) 복호화에 필요한 공개키 PK의 원소들(정확히는

$(a+2b+2)$개 중 $(2b-1)$개의 원소들이 복호화에 필요함)이 헤더에 포함되는 경우($b-HICBE_{BGW-1}$)와 2) 그 PK의 원소들이 비밀키에 포함되는 경우 ($b-HICBE_{BGW-2}$)로 나뉘어 진다. 특히 이는 통신량 또는 저장량이 중요한 사항으로 고려되는 응용환경에서 의미를 가질 수 있다.

일반적으로 브로드캐스트 암호의 효율성은 통신량, 저장량, 계산량으로 측정되는데, 전방향 안전성을 제공하는 공개키 브로드캐스트 암호의 특성을 반영하여 PK 사이즈와 비밀키 갱신에 필요한 계산량까지 비교 대상으로 한다. 여기서 통신량은 (R,ID,Hdr,C_M)에서 Hdr만 고려한다. [4]에서는 선택 평문 공격에 안전한 $b-HICBE_{BGW}$ 기법만 제시하고, 선택 암호문 공격에 안전한 기법은 명확하게 제시되지 않았다. 그러나 본 논문에서처럼 해쉬함수 기법의 아이디어를 적용하면 비슷하게 얻을 수 있으므로, (도출되는) 선택 암호문 공격에 안전한 $b-HICBE_{BGW}$ 기법을 가지고 효율성 비교를 한다. 구체적인 비교는 아래 표 1에 주어진다. 표 1에 나타난 효율성의 차이는 위에서 설명한 두 차이점들, 공개키 사이즈와 복호화 알고리즘의 입력값, 에 의해 생긴다.

표 1. $b-HICBE$ 기법의 효율성 비교

Table 1. Performance Comparison of $b-HICBE$ schemes for $n=ab$

	$b-HICBE_{BGW-1}$	$b-HICBE_{BGW-2}$	제안된 $b-HICBE$
통신량(Hdr)	$(a+2b+1)G''$	$(a+2)G''$	$(a+2)G''$
저장량	$(L+3)G''$	$(2b+L+2)G''$	$(b+L+3-r'')G'$
복호화 계산량	$3P+r'M+(L+6)E$	$3P+r'M+(L+6)E$	$3P+r'M+(L+6)E$
PK 사이즈	$(a+2b+2)G''$	$(a+2b+2)G''$	$(a+b+5)G''$
키갱신 계산량	$(L+2)E$	$(L+2)E$	$(L+2)E$

G'': G의 원소, P: G에서의 페어링, E: G에서의 지수승, M: G에서의 곱셈, $r'=\max\{r_1,\ldots,r_a\}$ (여기서 $r_i=|R_i|$임), $r''=\min\{r_1,\ldots,r_a\}$

표 1에서 통신량, 저장량, PK 사이즈는 그룹 G의 원소의 개수로서 표현하였다. 또한 저장량은 r'의 최대값을, 복호화 계산량은 r''의 최소값을 선택하였는데, 이는 worst case일 때의 성능을 나타내기 위함이다. 즉, 탈퇴자들의 부분집합 $R_1,\ldots,R_a$에 대한 각각의 저장량과 계산량 중 가장 좋지 않은 경우로 성능을 표현한다. 여기서 총 사용자 수 n은 고정된 값이고 $n=ab$이므로, a값이 커질수록 b값은 작아진다. 그러므로 a와 b가 포함된 통신량, 저장량, PK 사이즈는 a와 b의 선택에 의해 상관관계(tradeoff)를 가진다. 이 경우 a와 b의 선택은 통신량과 저장량 중 어느 것이 중요한지에 따라 구체적으로 결정될 것이다. 구체적으로 $b-HICBE_{BGW}$ 기법들과 제안된 기법의 효율성을 비교하면, 먼저 복호화와 키갱신에 필요한 계산량은 모두 동일하다. 다음으로 $b-HICBE_{BGW-2}$과 제안된 기법을 비교하면, 통신량 측면에서는 $(a+2)$개의 원소로 동일하나 사용자 기기에서의 저장량은 $(2b+L+2)<(b+L+3-r'')$이므로 제안된 기법이 더 적은 저장량을 가진다. 또한 이 두 기법은 $b-HICBE_{BGW-1}$에 비해 더 적은 통신량을 가지므로, 결과적으로 통신량이 중요한 환경에서는 본 논문에서 제안된 기법을 사용하는 것이 바람직하다. 또한 $b-HICBE_{BGW-1}$과 제안된 기법을 비교하면, 저장량 측면에서는 $b-HICBE_{BGW-1}$이 좋으나 통신량 측면에서는 제안된 기법이 좋은 것을 볼 수 있다. 그러므로 사용자 기기가 매우 제한된 저장용량을 가지는 환경이라면 (통신량을 증가시키면서) $b-HICBE_{BGW-1}$를 사용하는 것이 바람직할 것이다. 마지막으로 $a=b=\sqrt{n}$로 놓으면, 위의 세 $b-HICBE$ 기법들은 모두 $O(\sqrt{n})$ 사이즈의 통신량과 저장량을 갖는다.

V. 결 론

본 연구에서는 [5]에 기반하여 새로운 HICBE기법을 제안하였다. 제안된 HICBE기법은 ID 벡터를 시간에 대한 성분으로 바꾸고, $Derive$ 알고리즘을 통해 시간의 진화에 대응하도록 비밀키를 갱신하면 전방향 안전성(forward-secrecy)을 제공할 수 있다. 제안되는 HICBE기법은 기존의 [4]에서 BGW-PKBE기법에 기반한 HICBE기법과 유사한 성능을 보였다. 그러나 통신량이 중요한 응용환경에서는

[4]에서 제시된 기법보다 제안된 기법이 더 좋은 성능을 가진다. 이는 동일한 통신량에 비해 저장량이 더 적기 때문이며, 아울러 탈퇴자가 증가할수록 저장량이 줄어드는 장점도 가진다. 제안된 기법은 전체 사용자 수를 동일한 크기를 가지는 부분집합으로 분할함으로써, 공개키 사이즈와 저장량(비밀키 사이즈), 그리고 통신량에서 상관관계(tradeoff)를 제공할 수 있었다. 제안된 기법은 랜덤 오라클을 사용하지 않는 환경에서 BDHE가정을 사용하여 선택 암호문 공격에 안전하도록 증명되었다.

참 고 문 헌

[1] M. Naor and B. Pinkas, "Efficient trace and revoke schemes", Lecture Notes in Computer Science, vol. 1962, pp. 1-20, 2000.

[2] D. Boneh, C. Gentry, and B. Waters, "Collusion resistant broadcast encryption with short ciphertexts and private keys", Lecture Notes in Computer Science, vol. 3621, pp. 258-275, 2005.

[3] D. Yao, N. Fazio, Y. Dodis, and A. Lysyanskaya, "ID-based encryption for complex hierarchies with applications to forward security and broadcast encryption", ACM Press, pp. 354-363, 2004.

[4] N. Attrapadung, J. Furukawa, and H. Imai, "Forward-secure and searchable broadcast encryption with short ciphertexts and private keys", Lecture Notes in Computer Science, vol. 4284, pp. 161-177, 2006.

[5] D. Boneh, X. Boyen, and E. Goh, "Hierarchical identity based encryption with constant size ciphertext", Lecture Notes in Computer Science, vol. 3494, pp. 440-456, 2005.

[6] X. Boyen, Q. Mei, and B. Waters, "Direct chosen ciphertext security from identity-based techniques", ACM Press, pp. 320-329, 2005.

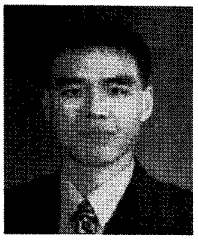
[7] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing", Lecture Notes in Computer Science, vol. 2139, pp. 213-229, 2001.

[8] C. Canetti, S. Halevi, and J. Katz, "Chosen ciphertext security from identity-based encryption", Lecture Notes in Computer Science, vol. 3027, pp. 207-222, 2004.

[9] D. Boneh and J. Katz, "Improved efficiency for cca-secure cryptosystems built using identity-based encryption", Lecture Notes in Computer Science, vol. 3376, pp. 87-103, 2005.

[10] D. Galindo and E. Kiltz, "Direct chosen ciphertext secure identity-based key encapsulation without random oracles", Lecture Notes in Computer Science, vol. 4058, pp. 336-347, 2006.

저 자 소 개



박 종 환

- 1999년 2월 : 고려대학교 수학과 (학사)
- 2004년 2월 : 고려대학교 정보보호대학원 정보보호학과 (석사)
- 2004년 3월 ~ 현재 : 고려대학교 정보경영공학전문대학원 정보보호학과 박사과정
- 주관심분야 : Pairing-based 암호, 브로드캐스트 암호, ID-based 암호, 전자서명 등

윤 석 구

- 1979년 2월 : 건국대학교 수학과 (학사)
- 1981년 2월 : 건국대학교 수학과 대학원 (석사)
- 1981년 ~ 2001년 : 국가정보원 정보보안단장
- 2002년 ~ 2004년 : 국가사이버안전센터장
- 2005년 2월 : 고려대학교 정보보호대학원 정보보호학과 (박사)
- 2005년 ~ 현재 : 고려대학교 정보경영공학전문대학원 BK21 연구교수
- 주관심분야 : 정보보호정책, 사이버 포렌식, 암호알고리즘 분석 등