情報保護學會論文誌
第17卷 第1號, 2007. 2

함축적인 인증을 제공하는 두 가지 공개키 암호 알고리즘의 안전성

박제홍,†‡ 이동훈, 박상우
국가보안기술연구소

Security of two public key schemes with implicit certifications

Je Hong Park,†‡ Dong Hoon Lee, Sangwoo Park
National Security Research Institute

요 약

본 논문에서는 ICISC 2004에 제안된 status certificate-based encryption(SCBE) 기법과 EUC Workshops 2006에 제안된 certificateless signature(CLS) 기법의 취약성을 제시한다. 이 두 기법은 ID 기반 암호시스템이 가지는 키 위탁 성질을 없애기 위해 사용자와 관리서버(CA 또는 KGC)가 함께 사용자 키(SCBE: 복호용 키, CLS: 서명용 키)를 생성하게 되어 있고, 이들 중 한쪽만을 제어할 수 있는 공격자는 암호시스템에 위해를 가할 수 있는 어떠한 이점도 가질 수 없다고 알려져 있다. 그러나, 본 논문에서는 각 기법의 공격모델에서 허용하는 공격자가 공개키 변조를 통해 독자적으로 사용자 키를 생성하여 사용할 수 있음을 보인다.

ABSTRACT

In this paper, we show that the status certificate-based encryption (SCBE) scheme proposed at ICISC 2004 and the certificateless signature (CLS) scheme proposed at EUC workshops 2006 are insecure. Both schemes are claimed that an adversary has no advantage if it controls only one of two participants making a cryptographic key such as a decryption key in SCBE or a signing key in CLS. But we will show that an adversary considered in the security model of each scheme can generate a valid cryptographic key by replacing the public key of a user.

Keywords : *Certificate-based Encryption, Certificateless signature, Security analysis*

Ⅰ. 서 론

ID 기반 암호시스템은 기존의 인증서 기반의 공개키 기반구조(Public key infrastructure: PKI)가 가지는 인증서 관리 측면의 어려움을 해결하기 위한 개념으로 제안되었다. ID 기반 암호시스템에서는 사용자의 식별자(Identifier)가 공개키로 직접 사용되고 TTP(Trusted Third Party)인 PKG(Private Key Generator)가 사용자 식별자(Identity: ID)로부터 개인키를 생성한다. 사용자는 이 개인키를 보안채널(secure channel)을 통해 PKG로부터 전달받아 사용하게 된다. 이러한 구조에서는 인증서 기반 PKI에서와 같은 인증서 검증을 통한 사용자의 명시적인 인증(explicit certification)은 불가능하지만, 해당 식별자를 가진 사용자만이 암호문의 복호화나 서명의 생성이 가능하게 하는 함축적인 인증(implicit certification)을 제공한다. ID 기반 암호시스템은 1984년 Shamir[1]에 의해 처음 제안되었고 키 공유(key

접수일: 2006년 9월 21일; 채택일: 2006년 11월 7일
† 주저자, 교신저자: jhpark@etri.re.kr

agreement)나 전자서명(digital signature)의 경우 다양한 방식이 제안된 바 있다. 반면 실제 사용가능한 암호기법(encryption scheme)의 설계는 미해결 문제로 남아 있었지만, 2001년 Boneh- Franklin[2]이 겹선형 함수를 이용한 암호기법을 제안하면서 ID 기반 암호시스템 자체가 새로운 연구주제로 주목받기 시작하였다.

이러한 연구와는 독립적으로, ID 기반 암호시스템과 같이 기존의 인증서 기반의 PKI와는 차별된 키 관리 체계를 가지는 암호시스템에 대한 연구도 진행되었다. 특히 이들의 주안점은 ID 기반 암호시스템의 안전성과 효율성은 그대로 유지하면서 PKG와 사용자 사이의 보안 채널과 키 위탁(key escrow)성질을 없애는 것이다. 이와 관련한 연구 결과들 중 대표적인 것으로 기존의 인증서 기반 PKI 구조를 간략화하는 인증서 기반 암호(certificate-based encryption: CBE)나 ID 기반 암호시스템처럼 아예 인증서를 사용하지 않는 무인증 공개키 암호시스템(certificateless public key cryptosystem: CL-PKC)을 들 수 있다. 두 개념은 서로 다른 키 관리 체계에서 키 위탁 성질을 없앨 수 있는 방법을 제시하고 있다.

1.1 Certificate-Based Encryption

2003년 Gentry에 의해 제안된 CBE[3]는 기존의 인증서 기반 PKI의 기본적인 체계는 그대로 유지한다. 하지만 ID 기반 암호(ID based encryption: IBE)에서와 같이 사용자에 대한 함축적 인증을 제공할 수 있는 방법을 도입하여 인증서 관리에 필요한 부담을 감소시키도록 한다. 구체적으로, 공개키 암호에서처럼 각 사용자는 자신의 공개키/개인키 쌍을 생성하고 CA에게 긴 주기를 가지는 인증서(long-lived certificate)를 요청한다. 그러면 CA는 IBE 기법을 사용하여 인증서 상태정보의 역할을 하는 부분복호용 키(partial decryption key)를 일정 시간 간격(time period)으로 생성한다. 즉, 각 시간 구간의 처음에 사용자 인증서의 상태가 유효한 경우에 한해서만 부분복호용 키를 생성하여 사용자에게 전송한다. CA로부터 부분복호용 키를 받은 사용자는, 자신의 개인키와 부분복호용 키를 합쳐서 복호용 키(decryption key)를 생성하게 된다. 이러한 구조를 통해 송신자 A가 수신자 B에 대한 사전 인증 없이 보낸 암호문은 B의 인증서가 유효한 상태에서만 복호화가 가능하게 된다. 이때 CA는 ID 기반 암호시스템의 PKG와는 달리 수신자의 개인키를 알 수 없기 때문에 암호문을 복호화할 수 없고, 부분복호용 키 전달 시 보안채널을 사용하지 않는다.

이러한 CBE의 구조적 특성은 인증서 폐지 문제(certificate revocation problem)에 대한 간단한 해결책을 제시하면 반면, 일정한 시간 간격에 따라 CA는 모든 사용자의 인증서 상태정보를 생성해서 전송해야 하기 때문에 그 계산 부담은 가중될 수 밖에 없다. 이러한 문제점을 해결할 수 있는 방법으로는 다중 CA(multiple CAs)를 두는 방법을 고려할 수 있겠지만[3], 다른 접근 방식으로 제안된 개념이 바로 상태인증서 기반 암호(status certificate-based encryption: SCBE)[4]이다.

SCBE에서 CA는 단지 사용자 공개키에 대한 인증을 제공하는 긴 주기를 가지는 인증서만을 발급하며, SCA (status certification authority)라고 하는 CA와 다른 기관에서 인증서의 현재 상태정보를 제공하는 짧은 주기(short-lived)의 상태인증서(status certificate)를 발급한다. 이때, CA가 발급하는 인증서는 사용자의 식별자와 사용자 공개키의 연관성을 보장하는 수단으로 사용자가 공개키에 대응하는 개인키를 안다는 것을 보장해준다. 반면 SCA가 발급하는 상태인증서는 사전에 정해진 시간에 대한 사용자 공개키의 유효성을 증명한다. 이러한 역할분담을 통해, CBE에서 CA가 부담하던 기능이 SCBE에서는 CA가 인증을 담당하고 SCA가 유효성을 담당하는 형태로 나눠지게 되지만, 기존 PKI에 대해 CBE가 가지던 장점은 SCBE에서도 여전히 유효하다.

1.2 Certificateless Public Key Cryptosystem

Al-Riyami와 Paterson에 의해 소개된 CL-PKC[5]는 ID 기반 암호시스템에서 인증서를 사용하지 않는 구조는 그대로 유지하면서 키 위탁 문제를 해결할 수 있는 방법을 제시하고 있다. ID 기반 암호시스템과 같이, 각 사용자는 유일한 식별자를 가지며, KGC(Key Generation Center)가 사용자 식별자와 연결되는 부분개인키(partial private key)를 생성한다. 하지만 ID 기반 암호시스템과 차별되는 점은, 이 부분개인키가 직접 사용자의 개인키로 사용되는 것이 아니라 CBE와 같이 사용자가 직접 생성하는 비밀정보와 합쳐져서 사용자 개인키[1]

1) 여기에서 사용자 개인키는 특정 암호 알고리즘에 따라 그 용도가 달라지게 된다. 예를 들어, 암호기법의 경우 CBE에서와 같이 복호용 키(decryption key)로 사용되고 서명기법의 경우에는 서명용 키(signing key)로 사용된다.

가 생성되는 것이다. 이를 통해 ID 기반 암호시스템이 가지는 키 위탁 성질이 없어지게 된다. 하지만 CBE와는 달리 KGC와 사용자 사이의 보안채널은 여전히 유효하다. 그리고 CL-PKC의 특징 중 하나는, 사용자 공개키가 사용자 식별자와는 독립적인 공개정보로 존재한다는 점이다. 이 공개키는 기존 인증서 기반 PKI에서처럼 신뢰받는 인증기관에 의해 인증된 값은 아니며, 단지 식별자와 함께 사용자 개인키와 연관된 공개정보로 활용된다.

1.3 본 논문의 결과

SCBE를 포함한 CBE나 CL-PKC는 앞에서 살펴본 바와 같이 키 위탁 성질을 없애기 위해 관리서버(CA 또는 KGC)와 사용자가 각자 생성한 비밀 정보를 결합하여 암호 알고리즘에 적용되는 사용자 비밀키(CBE의 경우 복호용 키, CL-PKC의 경우 사용자 개인키)를 생성한다[2]. 그러므로 CBE 기법이나 CL-PKC는 사용자나 관리서버 중 한쪽을 제어할 수 있는 공격자가 다른 한쪽을 제어할 수 없는 이상 암호시스템에 위해를 가하는 것이 불가능하도록 설계되어야 한다. 하지만 본 논문에서는 Yum과 Lee가 ICISC 2004에서 제안한 SCBE 기법(이하 Yum-Lee SCBE 기법)[4]과 Yap, Heng 그리고 Goi가 EUC Workshops 2006에서 제안한 무인증 서명(certificateless signature: CLS) 기법(이하 YHG CLS 기법)[6]이 공개키 변조 공격(key replacement attack)에 취약함을 보인다. 본 논문에서의 공격은 두 기법에서 사용하는 사용자 비밀키가 가지는 공통적인 대수적 구조에 기인한다. 두 기법 모두 사용자 비밀키는 관리서버와 사용자가 공통으로 생성하는 사용자 식별자에 대한 BLS 다중서명(multisignature)[7]의 형태를 가지고 있다. 참고로 사용자 비밀키를 BLS 서명으로 보는 관점은 이미 Boneh-Franklin의 ID 기반 암호기법[2]에 적용된 바 있으며, Gentry의 CBE 기법[3]에서는 사용자 비밀키인 복호용 키를 다중서명의 일종인 BGLS aggregate 서명[8]의 형태로 생성한다. 일반적으로 BLS 다중서명에 대해서는 다음과 같은 내부자에 의한 공개키 변조 공격(key replacement attack 또는 rogue attack)이 가능하다.[7,8] 두 사용자 A와 B의 공개키, 개인키 쌍을 각각 $(g^{x_A}, x_A), (g^{x_B}, x_B)$, 메시지를 m, 그리고 해쉬함수를 H라 하자. B는 서명 생성 이전에 자신의 공개키 g^{x_B}를 g^{x_B}/g^{x_A}로 교체한다. 그러면 B는 m에 대해

$$H(m)^{x_B} \quad (= H(m)^{x_A} \cdot H(m)^{x_B - x_A})$$

를 계산한 후 이 값을 A와 함께 생성한 다중서명이라고 주장한다. 서명 검증자는 관계식

$$e(H(m), g^{x_A} \cdot (g^{x_B}/g^{x_A})) = e(H(m)^{x_B}, g)$$

을 확인함으로써, 이 서명 생성에 A와 B가 모두 참여한 것으로 판단하게 된다. 본 논문에서 제시하는 Yum-Lee SCBE 기법과 YHG CLS 기법에 대한 공개키 변조 공격은 기본적으로 이러한 BLS 다중서명에 대한 공개키 변조 공격을 이용하며, 자세한 내용은 다음과 같다.

Yum-Lee SCBE 기법에서 고려하는 두 가지 형태의 공격자는 모두 사용자 비밀키인 복호용 키의 생성에 관여하고 있다. 여기에서 복호용 키는 SCA의 개인키에 의해 생성된 BLS 서명인 부분복호용 키와 사용자의 개인키에 의해 생성된 BLS 서명을 합친 BLS 다중서명으로 볼 수 있다. 그러므로 공격자는 상기 BLS 다중서명에 대한 공개키 변조 공격을 적용하여 복호용 키가 관리서버와 사용자가 협조하지 않고 독자적으로 생성될 수 있음을 보인다.

또한 YHC CLS 기법의 경우에는 사용자의 비밀키인 개인키 생성과 관계없는 제 3자가 개인키를 위조하여 서명을 생성할 수 있음을 보인다. 공격자는 상기 BLS 다중서명에 대한 공개키 변조 공격을 이용하여 서명자의 개인키를 임의로 생성한 후, 이를 이용하여 YHG 서명을 위조한다. 앞에서 언급한 바와 같이 다중서명에 대한 공개키 변조 공격은 서명에 참여하는 내부자에 의해 이뤄진다. 그러나 CL-PKC에서는 특정 사용자의 키를 임의로 변경할 수 있는 공격자를 공격 모델의 하나로 가정하기 때문에 개인키 생성에 참여하지 않은 제 3자에 의한 서명 위조가 가능하게 된다. 특히 이러한 공격은 YHG CLS 기법에서 개인키가 제 3자에 의해 임의로 조작될 수 있음을 보여준다.

1.4 기존 결과

2) CBE와 CL-PKC에서는 비슷한 역할을 하는 값들이 다른 용어로 표현되어 있다. CBE에서 인증서 발급에 사용되는 공개키에 대응하는 사용자 고유의 값을 개인키라 하고 CA와 사용자가 같이 만든 비밀값을 그 용도에 한정해서 복호용 키라 부른다. 반면 CL-PKC에서는 CBE의 개인키가 사용자 비밀값에 해당되고 KGC와 사용자가 같이 만든 비밀값을 사용자 개인키라 한다.

CBE나 CL-PKC는 2003년 개념이 제안된 이후 다양한 기법들이 제안되고 있다. CBE의 경우 그 구조적 특성에서 보듯 IBE나 CLE(certificateless public key encryption)를 이용한 generic conversion 방식이 제안되었지만,[9,10] 이들 기법들은 모두 CBE의 안전성 모델에 부합하지 못함이 밝혀졌다.[11,12] CL-PKC의 경우 암호와 서명 각각 다양한 방식이 제안된 바 있으나 여기에서는 서명에 대한 결과만 간략히 소개하며 암호에 대해서는 논문[13]을 참조하기 바란다. 서명의 경우 몇 가지 제안된 기법[5,14,15]들에 대해 최근 다양한 취약성이 제시되고 있다.[16,17,18] 특히 Gorantla-Saxena CLS 기법의 경우 본 논문에서 다루는 YHG CLS 기법과 마찬가지로 서명 검증 시 서명자 공개키와 서명이 동시에 검증되지 않기 때문에 공개키 변조 공격이 성립하였다.[18]

5. 겹선형 함수

겹선형 함수(bilinear map) $e: G_1 \times G_1 \rightarrow G_2$는 효율적인 계산이 가능하고, $e(g^r, g^s) = e(g,g)^{rs}$인 겹선형성을 만족하는 사상이다. 여기에서 G_1과 G_2는 모두 소수 위수(order) p를 가지는 곱셈군으로 각각 $g(\neq 1)$와 $e(g,g)$에 의해 생성된다. 참고로 겹선형 함수를 실제 구현할 수 있는 경우는 초특이 곡선(supersingular curve)으로 대표되는 특수한 성질을 가지는 (초)타원곡선군에서 정의되는 Weil/Tate pairing을 이용하는 경우밖에 없다. 그러므로 논문에서 겹선형 함수는 일반적으로 pairing과 거의 동등한 의미로 사용되며 아래 YHG CLS 기법의 경우에도 [6]에서는 G_1을 덧셈에 의해 정의되는 타원곡선군으로 기술하고 있다. 그러나 본 논문에서는 원 논문의 내용을 해치지 않는 범위 내에서 SCBE 기법과의 표현을 일치시키기 위해 곱셈군을 가정하여 설명하도록 한다.

Ⅱ. Yum-Lee SCBE 기법에 대한 안전성

2.1 Yum-Lee SCBE 기법

다음 해쉬함수 $H_1: \{0,1\}^* \rightarrow G_1$, $H_2: G_1 \rightarrow \{0,1\}^n$, $H_3: \{0,1\}^n \times \{0,1\}^n \rightarrow Z_p^*$, $H_4: \{0,1\}^n \rightarrow \{0,1\}^n$는 겹선형 함수와 함께 CA가 생성하는 시스템 파라미터에 포함되는 정보이다. Yum-Lee SCBE 기법은 다음과 같이 설명할 수 있다.

- CA 파라미터 생성: 주어진 안전성 파라미터(security parameter) k에 대해, CA는 $s_{CA} \in Z_p^*$를 임의로 선택하고 $u_{CA} = g^{s_{CA}} \in G_1$를 계산한다. CA는 마스터키 $mk_{CA} = s_{CA}$를 비밀로 보관하고, 시스템 파라미터 $params = \langle G_1, G_2, e, g, u_{CA}, H_1, H_2, H_3, H_4, p, n \rangle$을 공개한다.
- SCA 파라미터 생성: 주어진 시스템 파라미터 $params$에 대해, SCA는 임의로 $s_{SCA} \in Z_p^*$를 생성하고 $g^{s_{SCA}} \in G_1$를 계산한다. SCA의 공개키 pk_{SCA}와 마스터키 mk_{SCA}는 각각 다음과 같다.

$$pk_{SCA} = g^{s_{SCA}},\ mk_{SCA} = s_{SCA}$$

- 사용자 키 설정: 주어진 $params$와 사용자 식별자 id에 대해, 사용자는 임의로 $s_{id} \in Z_p^*$를 선택하고 $g^{s_{id}}$를 계산한다. 사용자의 공개키 pk_{id}와 개인키 sk_{id}는 각각 다음과 같다.

$$pk_{id} = g^{s_{id}},\ sk_{id} = s_{id}$$

- CA 인증서 발급: 주어진 $params$, 사용자 식별자 id, 그리고 공개키 pk_{id}에 대해, CA는 사용자 id의 정보 λ를 검증하고 그 결과가 유효할 경우 인증서 $Cert_{id} = g_{id}^{s_{CA}} \in G_1$를 계산하여 사용자 id에게 전송한다. 이때 $g_{id} = H_1(id, pk_{id}, params, \lambda)$.
- SCA 인증서 갱신: SCA는 시간 t에 대한 사용자 id의 인증서 $Cert_{id}$의 폐지여부를 확인한 후, 아직 폐지되지 않았을 경우 상태인증서 $SCert_{(id,t)} = g_{(id,t)}^{s_{SCA}} \in G_1$를 계산하여 사용자 id에게 전송한다. 이때 $g_{(id,t)} = H_1(id, t, pk_{id}, params, \lambda)$.
- 사용자 갱신: 주어진 $SCert_{(id,t)}$에 대해, 시간 t의 복호용 키 $dk_{(id,t)} = g_{(id,t)}^{s_{id}} \cdot SCert_{(id,t)} \in G_1$를 계산한다.
- 암호화: 메시지 m을 암호화하기 위해, 우선 임의로 $\alpha \in \{0,1\}^n$을 선택하고 $r = H_3(\alpha, m)$을 구한다. 그러면 m에 대한 암호문 C는 다음과 같이 계산한다.

$$C = (g^r, \alpha \oplus H_2(e(pk_{SCA} \cdot pk_{id}, g_{(id,t)})^r), m \oplus H_4(\alpha))$$

- 복호화 : 주어진 암호문 $C = (c_1, c_2, c_3)$에 대해, 먼저 $\alpha = c_2 \oplus H_2(e(c_1, dk_{(id,t)}))$를 계산한 후 $m = c_3 \oplus H_4(\alpha)$와 $r = H_3(\alpha, m)$을 차례로 계산한다. 만일 $c_1 = g^r$이 사실이면 m을 반환하고, 아니면 $\perp$를 반환한다.

CBE[3]나 SCBE[4]에서 사용하는 (상태)인증서는 일반적인 PKI에서 사용하는 인증서와는 그 모양이나 기

능이 틀리지만, 본 논문에서는 논문 [3,4]의 기술을 그대로 따른다. CBE와는 다르게 SCA라는 새로운 구성원이 있기 때문에 SCBE에 대한 안전성 모델은 CBE에 비해 복잡하다. 논문 [4]에 제시된 안전성 모델에서는 CBE처럼 2개가 아닌 3개의 다른 공격 모델을 정의하고 있으며, 각각의 공격 모델에서 공격자는 노출된 CA (eavesdropping CA), 인증되지 않은 사용자(uncertified user), 그리고 정직하지 않은 SCA(dishonest SCA)의 역할을 가정한다. 하지만 논문 [4]에서는 실제 제안한 SCBE 기법의 안전성 증명을 위해 인증받지 않은 사용자와 노출된 CA를 하나의 공격자로 보고, SCBE의 안전성 모델을 두 가지 공격모델을 가지는 형태로 변형시켰다. 이 변형된 안전성 모델에서는 I형 공격자 A_I가 노출된 CA와 인증받지 않은 사용자의 역할을 동시에 수행하고, II형 공격자 A_{II}는 정직하지 않은 SCA의 역할을 수행한다. 이러한 변형된 안전성 모델은 그 구조가 CBE의 그것과 유사한 것으로 보이지만, SCBE의 안전성 모델에서는 CA가 인증받지 않은 사용자와 협력하는 능동적 공격자(active attacker)가 될 수 있다는 차이를 가진다.

2.2 안전성 분석

Yum-Lee SCBE 기법에서, 사용자는 시간 t에서만 유효한 복호용 키 $dk_{(id,t)}$를 SCA와 사용자가 참여한 BLS 다중서명

$$g_{(id,t)}^{s_{id}} \cdot SCert_{(id,t)} = g_{(id,t)}^{s_{id}} \cdot g_{(id,t)}^{s_{SCA}}$$

의 형태로 계산한다. 하지만 BLS 다중서명의 대수적 구조에 의해, 공개키 대치를 이용한 I형과 II형 공격자의 위조 공격을 적용할 수 있다.

우선 I형 공격자 A_I의 경우를 살펴보자. A_I는 인증받지 않은 사용자 id의 공개키 pk_{id}를 pk_{id}/pk_{SCA}로 대치한다. 비록 A_I는 pk_{id}/pk_{SCA}의 이산대수는 모르지만 $g'_{(id,t)} = H_1(id,t,pk_{id}/pk_{SCA},params,\lambda)$를 이용하여 $g'^{s_{id}}_{(id,t)}$를 사용자 id의 복호용 키 $dk'_{(id,t)}$로 사용할 수 있다. 즉, A_I는 SCA의 상태인증서 유무에 관계없이 독자적으로 복호용 키를 생성할 수 있게 된다. 기본적으로 SCBE에서 SCA는 사용자의 long-term 인증서를 기반으로 사용자 공개키의 폐지 여부만 확인하므로, CA의 마스터키를 알고 있는 A_I는 변경된 공개키에 대한 인증서를 발급하고 SCA가 이에 대한 상태인증서를 발급하게 할 수 있다. 사용자 id에게 메시지 m을 암호화해서 보내려는 송신자는 id의 공개키 pk_{id}/pk_{SCA}를 사용하여 암호문 $C=(c_1,c_2,c_3)$를 계산한다. 여기에서 $c_2 = \alpha \oplus H_2(e(pk_{SCA} \cdot pk_{id}/pk_{SCA}, g'_{(id,t)})^r)$이며, 이때 주의할 점은 SCBE의 특성상 송신자는 사용자 id의 상태인증서를 확인하지 않는다는 것이다. 공격자 A_I는 암호문으로부터 얻게되는 관계식

$$\begin{aligned} c_2 &= \alpha \oplus H_2(e(pk_{SCA} \cdot pk_{id}/pk_{SCA}, g'_{(id,t)})^r) \\ &= \alpha \oplus H_2(e(g^{s_{id}}, g'_{(id,t)})^r) \\ &= \alpha \oplus H_2(e(g^r, g'^{s_{id}}_{(id,t)})) \\ &= \alpha \oplus H_2(e(c_1, dk'_{(id,t)})) \end{aligned}$$

으로부터 α를 알 수 있으므로, m을 얻을 수 있다.

이와 비슷하게, II형 공격자 A_{II}는 SCA의 공개키 pk_{SCA}를 특정 사용자 id에 대해 pk_{SCA}/pk_{id}로 바꿔서 공개한다. A_{II}는 pk_{SCA}/pk_{id}의 이산대수는 모르지만 $g_{(id,t)}^{s_{SCA}}$를 특정 시간 t에서의 사용자 id의 복호용 키로 사용할 수 있다. 즉, A_{II}는 사용자의 개인키를 모르는 상태에서도 사용자의 복호용 키를 생성할 수 있다. 물론 이러한 공격은 공격자가 인증서 발급을 담당하는 CA의 마스터키에 접근할 수 없기 때문에 A_I의 공격보다는 제약이 따를 수 밖에 없지만, SCA가 CA로부터 인증서를 발급받지 않거나 사용자에 의한 SCA 공개키의 명시적인 인증절차가 없는 시스템 운용의 경우에 대한 일반적인 공격으로 충분히 가능하다.

결론적으로 이러한 공격은 SCA 또는 사용자가 서로 협력하지 않고 독자적으로 복호용 키를 제어할 수 있음을 의미한다. 다중서명의 경우 이러한 형태의 공격을 막기 위해, 서명생성에 참여하는 각 사용자들이 CA에게 공개키를 등록할 때 자신의 공개키에 대응하는 개인키를 알고 있음을 증명할 것을 요구한다.[7,8] 그러나 Yum-Lee SCBE 기법에 대해서는 이러한 방법이 성립하지 않는다. 이는 I형 공격자가 노출된 CA를 가정하기 때문이다. 하지만 같은 메시지를 서명하는 다중서명과는 달리, 다른 메시지에 대한 서명을 생성하여 하나의 서명으로 표현하는 aggregate 서명의 형태를 복호용 키 생성에 사용함으로써 키 대치 공격을 막을 수 있다. 이는 Gentry의 CBE 기법에 적용된 바 있다.[3]

Ⅲ. YHG CLS 기법에 대한 안전성

3.1 YHG CLS 기법

다음 $H_1 : \{0,1\}^* \rightarrow G_1$과 $H_2 : \{0,1\}^* \times G_1 \rightarrow Z_p^*$는 해쉬함수로, 겹선형 함수 정보와 함께 KGC에 의해 생성되는 시스템 파라미터의 일부로 사용된다. YHG CLS 기법은 다음과 같다.

- KGC 파라미터 생성: 주어진 안전성 파라미터(security parameter) k에 대해, KGC는 임의의 생성자 $g \in G_1$을 선택하고 $s \in Z_p^*$를 임의로 선택한 후, $g_0 = g^s \in G_1$로 둔다. 이어 시스템 파라미터로 $params = \langle G_1, G_2, e, g, g_0, H_1, H_2, p \rangle$을 공개하고 마스터키 $mk = s$는 비밀로 보호한다.
- KGC 부분개인키 생성: 주어진 $params$과 mk에 대해, KGC는 $h_A = H_1(id_A) \in G_1$을 계산하고 부분개인키 $d_A = h_A^s$를 출력한다.
- 사용자 비밀값 생성: 주어진 $params$에 대해, 사용자 A는 임의의 값 $x_A \in Z_p^*$를 사용자 비밀값으로 선택한다.
- 사용자 개인키 생성: 주어진 부분개인키 d_A에 대해, 사용자 A는 자신의 개인키 $s_A = h_A^{x_A} \cdot d_A \in G_1$를 계산한다.
- 사용자 공개키 생성: 주어진 $params$과 비밀값 x_A에 대해, 사용자 A는 $g_A = g^{x_A} \in G_1$을 계산하여 자신의 공개키로 공개한다.
- 서명 생성: 주어진 $params$, id, s_A, 그리고 메시지 m에 대해, 사용자 A는 임의로 $a \in Z_q^*$를 선택하고 $u = h_A^a \in G_1$을 계산한다. 다음에는 $h = H_2(m,u) \in Z_p^*$와 $v = s_A^{(a+r)}$를 차례로 계산하여 $\sigma = (u,v)$를 m에 대한 서명으로 출력한다.
- 서명 검증: 주어진 서명/메시지 쌍 (σ, m), 서명자의 식별자 id_A, 그리고 서명자의 공개키 g_A에 대해, 검증자는 먼저 $r = H_2(m,u)$를 계산하고 $e(g,v) = e(g_0 \cdot g_A, u + h_A^s)$인지 확인한다. 틀리면 서명을 거부(reject)하고 맞으면 승인(accept)한다.

저자들은 서명 검증단계에서 기존의 서명기법에 비해 적은수의 겹선형 함수 계산을 요구하므로 효율적이라는 주장을 하였다. 위에서 설명한 바와 같이, YHG 서명기법은 서명 검증단계에서 2번의 겹선형 함수 계산을 필요로 한다. 구체적으로 보면, 이러한 서명기법의 효율성은 서명 검증단계에서 독립적으로 실시하는 서명자에 대한 공개키 유효성 검증을 생략했기 때문에 가능한 것으로 아래에서 이러한 생략에 의해 공개키 변조 공격에 취약함을 증명한다.

3.2 안전성 분석

일반적으로, 서명자는 서명 생성 시 의도한 검증자에게 서명과 함께 자신의 식별자와 공개키를 같이 전달하게 된다. 그러므로 식별자 id_A를 가진 사용자 A의 서명을 위조하기 위해 공격자는 아래와 같은 행동을 한다.

- 임의로 $x \in Z_p^*$를 선택하고 메시지 m에 대한 서명 $\sigma = (u,v)$를 다음과 같이 생성

$$u = h_A^a, r = H_2(m,u), v = h_A^{x(a+r)}$$

- 사용자 A의 공개키 g_A를 $g'_A = g^x / g_0$로 변경, 이어서 서명 $\sigma = (u,v)$, 메시지 m, 식별자 id_A, 그리고 공개키 g'_A을 검증자에게 송부

그러면 검증자는 $h = H_2(m,u)$를 계산하고

$$\begin{aligned} e(g,v) &= e(g, h_A^{x(a+r)}) \\ &= e(g^x, h_A^{(a+r)}) = e(g_0 \cdot g'_A, u \cdot h_A^r) \end{aligned}$$

를 확인하게 되어, 결국 σ를 메시지 m에 대해 A가 생성한 유효한 서명으로 인식하게 된다.

결국 이러한 위조공격은 공격자가 사용자 A의 공개키를 g^x / g_0로 위조한 후 h_A^x 자체를 서명용 키로 사용하는 것으로, 앞에서 기술한 BLS 대리서명에 대한 공개키 변조 공격 방식을 통해 CLS 기법에서 사용하는 서명용 개인키를 공격자 임의로 만들어낼 수 있음을 보여주는 것이다. 앞 절의 Yum-Lee SCBE 기법에 대한 공격과는 달리, YHG CLS 기법의 경우 서명용 개인키 생성과는 관계없는 제 3자를 공격자로 가정한다. 그러나 CL-PKC의 공격모델에서는 사용자의 공개키에 대한 별도의 인증 장치가 없음을 반영하여 KGC를 제외한 공격자가 특정 사용자의 키를 임의로 변경하는 것을 허용한다. 그러므로 Yum-Lee SCBE 기법에 대한 공격과 마찬가지로 공개키 대치 공격 방식을 적용할 수 있다.

CLS에서 이러한 형태의 공격에 안전하기 위해서는 서명 검증단계에서 서명자가 자신의 공개키에 대응하는 비밀값을 알고 있음을 증명하는 과정이 필요하다. YHG CLS 기법에서 이를 제공하기 위한 한 가지 방법으로는 우선 사용자 A의 공개키 $g_A = g^{x_A}$에 $g_0^{x_A}$를 추가하고 공개키 유효성 검증식

$$e(g_0, g_A) = e(g, g_0^{x_A}) \qquad (1)$$

을 서명 검증과정에 추가하는 것을 고려할 수 있다. 이 식은 기본적으로 서명자 A의 공개키 $\langle X, Y\rangle$가 $Y = sX$의 관계를 만족하는지 확인시켜 준다. 여기에서 $Y = g_0^{x_A}$ 그리고 $X = g^{x_A}$. 또한 만일 공격자가 A의 공개키 g_A를 $g'_A = g^x / g_0$의 형태로 변형할 경우 식 (1)을 통과할 수 없기 때문에, x_A가 정확하게 개인키 생성에 사용되었는지도 같이 확인할 수 있게 된다. 물론 이러한 절차를 포함시킴으로써 2번이 아닌 4번의 겹선형 함수 연산이 필요하게 되므로 (물론 같은 사용자가 생성한 다수의 서명을 검증할 경우에는 처음에만 공개키 유효성 검증을 하면 되므로 2번으로 줄 수 있다.), 기존 서명기법에 대한 효율성 이점은 사라지게 된다.

Ⅳ. 결 론

사용자 키에 대한 명시적인 인증을 사용하지 않는 CBE나 CL-PKC는 기존의 인증서 기반 암호시스템에 비해 암호기법 설계 시 키의 생성에 있어서 많은 주의를 필요로 한다. 본 논문에서는 BLS 다중서명에 대한 공개키 대치 공격 방식을 이용하여 ICISC 2004에서 Yum-Lee가 제안한 SCBE 기법과 Yap-Heng-Goi가 EUC Workshops 2006에 제안한 CLS 기법에 대한 취약성을 제시하였다. 이는 두 기법에서 복호용/서명용 키의 형태가 두 참여자에 의해 생성되는 BLS 다중서명의 형태를 가진다는 특징을 이용한 것이다. 본 논문에서는 이러한 공격에 대한 대응 방법을 간략하게 소개하였지만, 이는 제시된 알고리즘에 의존하는 것으로 새로운 암호기법 설계 시 이러한 취약성에 항상 주의할 필요가 있다.

참고문헌

[1] A. Shamir, "Identity-based cryptosystems and signature schemes," Advances in Cryptology - CRYPTO 1984, LNCS 196, pp. 47-53, 1985.

[2] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," SIAM J. Comput., vol. 32, no. 3, pp. 586-615, 2003.

[3] C. Gentry, "Certificate-based encryption and the certificate revocation problem," Advances in Cryptology-EUROCRYPT 2003, LNCS 2656, pp. 272-293, 2003.

[4] D.H. Yum and P.J. Lee, "Separable implicit certificate revocation," Information Security and Cryptology - ICISC 2004, LNCS 3506, pp. 121-136, 2005.

[5] S.S. Al-Riyami and K.G. Paterson, "Certificateless public key cryptography," Advances in Cryptology-ASIACRYPT 2003, LNCS 2894, pp. 452-473, 2003.

[6] W.-S. Yap, S.-H. Heng and B.-M. Goi, "An efficient certificateless signature scheme," EUC Workshops 2006, LNCS 4097, pp. 322-331, 2006.

[7] A. Boldyreva, "Efficient threshold signature, multisignature and blind signature schemes based on the gap-Diffie-Hellman-group signature scheme," Public Key Cryptography-PKC 2003, LNCS 2567, pp. 31-46, 2003.

[8] D. Boneh, C. Gentry, B. Lynn and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," Advances in Cryptology-EUROCRYPT 2003, LNCS 2656, pp. 416-432, 2003.

[9] D.H. Yum and P.J. Lee, "Identity-based cryptography in public key management," Public Key Infrastructure-EuroPKI 2004, LNCS 3093, pp. 71-84, 2004.

[10] S.S. Al-Riyami and K.G. Paterson, "CBE from CL-PKE: A generic construction and efficient scheme," Public Key Cryptography-PKC 2005, LNCS 3386, pp. 398-415, 2005.

[11] B.G. Kang and J.H. Park, "Is it possible to have CBE from CL-PKE?" Cryptology ePrint Archive, Report 2005/431.

[12] D. Galindo, P. Morillo and C. Rafols, "Breaking Yum and Lee generic construction of certificate-less and certificate-based encryption schemes," Public Key Infrastructure-EuroPKI 2006, LNCS 4043, pp. 81-91, 2006.

[13] A.W. Dent, "A survey of certificateless encryption schemes and security models," Cryptology ePrint Archive, Report 2006/211.

[14] D.H. Yum and P.J. Lee, "Generic construction

of certificateless signature," Information Security and Privacy-ACISP 2004, LNCS 3108, pp. 188-199, 2004.

[15] M.C. Gorantla and A. Saxena, "An efficient certificateless signature scheme," Computational Intelligence and Security-CIS 2005, LNAI 3802, pp. 110-116, 2005.

[16] X. Huang, W. Susilo, Y. Mu and F. Zhang, "On the security of certificateless signature schemes from ASIACRYPT 2003," Cryptology and Network Security-CANS 2005, LNCS 3810, pp. 13-25, 2005.

[17] B.C. Hu, D.S. Wong, Z. Zhang and X. Deng, "Key replacement attack against a generic construction of certificateless signature," Information Security and Privacy-ACISP 2006, LNCS 4058, pp. 235-246, 2006.

[18] X. Cao, K.G. Paterson and W. Kou, "An attack on a certificateless signature scheme," Cryptology ePrint Archive, Report 2006/367.

〈著者紹介〉

박 제 홍 (Je Hong Park) 정회원
1998년 2월: 경북대학교 수학과 졸업
2000년 2월: 한국과학기술원 수학과 석사
2004년 2월: 한국과학기술원 수학과 박사
2004년 3월~현재: 국가보안기술연구소 연구원
<관심분야> 암호론, 응용정수론

이 동 훈 (Dong Hoon Lee) 정회원
1994년 2월: 서울대학교 수학교육과 졸업
1996년 2월: 한국과학기술원 수학과 석사
2000년 2월: 한국과학기술원 수학과 박사
2000년 2월~2002년 3월: (주)퓨쳐시스템 선임연구원
2002년 4월~현재: 국가보안기술연구소 선임연구원
<관심분야> 응용정수론, 암호론, 인터넷보안

박 상 우 (Sangwoo Park) 정회원
1989년 2월: 고려대학교 수학교육과 졸업
1991년 8월: 고려대학교 수학과 석사
2003년 2월: 고려대학교 수학과 박사
1991년 8월~1999년 12월: 한국전자통신연구원 선임연구원
2000년 1월~현재: 국가보안기술연구소 책임연구원
<관심분야> 암호론, 정보보호