情報保護學會論文誌
第 22 卷 第 6 號, 2012. 12

익명성을 보장하는 비대칭 공모자 추적 기법의 설계

이 문 식,[1†] 강 순 부,[1‡] 이 주 희[2]
[1]공군사관학교, [2]국가수리과학연구소

Construction of an Asymmetric Traitor Tracing Schemes with Anonymity

Moonsik Lee,[1†] SunBu Kang,[1‡] Juhee Lee[2]
[1]Korea Air Force Academy, [2]National Institute for Mathematical Sciences

요 약

공모자 추적 기법은 불법 디코더를 만드는데 관여한 사용자(공모자)들 중에서 적어도 한명의 사용자를 추적함으로서 사용자의 개인키를 악의적인 목적으로 공유하지 못하게 하는 기법이다. 일반적으로 공모자 추적 기법에서는 시스템 매니저가 모든 사용자들의 개인키를 생성하여 배분하는 것이라 가정하지만, 시스템 매니저가 모든 사용자들의 개인키를 알고 있다면 불법 디코더가 발견되었을 때, 이를 만들기 위해 어떤 사용자가 공모했다는 사실을 제3자에게 확신시킬 수 없는 단점이 있다. 시스템 매니저가 모든 사용자의 개인키를 알 수 없도록 설계하여 그러한 단점을 해결할 수 있고 나아가 개인 프라이버시를 증대시키기 위한 기법이 비대칭 공모자 추적 기법이다. 기존에 두편의 비대칭 공모자 추적 기법들이 제안되었지만 한편은 안전하지 않다는 것이 증명되었다. 본 논문에서는 다른 한편의 기법이 안전하지 않다는 것을 언급하고, 공모자 추적 기법과 익명성의 연결성을 연구하여, 이를 바탕으로 실질적이고 새로운 기법을 제안하고자 한다. 제안하는 기법은 익명 인증 시스템과 비대칭 공모자 추적 기법을 연결하는 구체적인 기법으로 안전성을 증명하고 적용할 수 있는 응용분야를 함께 제시하고자 한다.

ABSTRACT

Traitor tracing schemes deter traitors from sharing their private keys by tracing at least one of the subscribers who were implicated in the construction of a pirate decoder. In general, it is assumed that the system manager in the scheme generates and distributes the subscribers' private key. But if the system manager knows the subscribers' private keys, he cannot convince a third party of a certain subscriber's piracy. To solve this problem, the system manager should not know the whole parts of subscribers' private keys and this leads to researches of asymmetric schemes. Moreover for the purpose of enhancing subscribers' privacy, there were two proposals of introducing anonymity onto asymmetric traitor tracing schemes, but one of them turned out to be a failure. In this paper, we point out that the other proposal also has flaws. We consider how to introduce anonymity to traitor tracing schemes, as a result, we suggest a new framework which is practical. We also construct a scheme by using an anonymous credential system and an asymmetric traitor tracing scheme. We prove the security of our scheme and consider the typical applications.

Keywords: Broadcast encryption, Traitor Tracing, Anonymity

I. Introduction

Traitor tracing schemes are broadcast encryption systems where at least one of the traitors who were implicated in the

접수일(2012년 3월 8일), 수정일(1차: 2012년 9월 6일), 게재확정일(2012년 10월 12일)
† 주저자, kafa04@snu.ac.kr
‡ 교신저자, sbkang@postech.ac.kr

construction of a pirate decoder can be traced. This traceability is required in various contents delivery system like satellite broadcast, DMB, pay-TV, DVD, online database and so on. In general, traitor tracing schemes were introduced in symmetric setting, where the subscribers share all of their secret information with the system manager. A shortcoming of this setting is that the system manager cannot obtain undeniable proof of the implication of certain subscribers in the construction of a pirate decoder. Furthermore, if the system manager is malicious, he can implicate an innocent subscriber in the piracy.

In 1997, Pfitzmann[1] pointed out this problem and introduced an asymmetric traitor tracing scheme using an interactive key distribution protocol. In the asymmetric scheme, the system manager cannot construct a pirate decoder to frame an innocent subscriber. Since then, asymmetric traitor tracing schemes have been researched in [2-5]. However, the scheme of [4] turned out to be a symmetric one and it was pointed out that the schemes of [3,5] had flaws by Kiayias and Yung[2], who suggested a new scheme.

In 2001, to enhance the subscribers' privacy further, a concept of anonymous asymmetric scheme, in which subscribers hide their identities from the system manager, was proposed in [6]. However this scheme failed to achieve such anonymity. As a successive research, in 2003, Choi et al.[7] proposed two anonymous asymmetric schemes by introducing a trusted agent and pseudo identities.

As a way of protecting one's privacy, an anonymous credential system, where a user can prove his various credentials with no leakage of identity information, was introduced. Camenisch and Lysyanskaya[8] proposed secure signature schemes using bilinear maps, each of which can be used to compose an anonymous credential system.

In this paper, we show that the schemes of [7] have flaws. One of their schemes is based on the one-time scheme of Kurosawa and Desmedt[4] and the other is on the asymmetric public-key scheme[2]. But the first scheme is insecure since a linear attack is possible and the second scheme is designed on the same framework of the first scheme so it is too over-loaded to use in practice. After analyzing the framework of them, we suggest a new framework for constructing anonymity with traitor tracing schemes. As a concrete scheme, we construct an anonymous asymmetric traitor tracing scheme using an anonymous credential system and an asymmetric traitor tracing scheme. To raise the efficiency we modify the schemes and prove the security. We show two practical applications where both privacy and traitor tracing are important.

II. Preliminaries

In this Section we describe some preliminaries to understand our paper. To avoid messy pieces, we omit the descriptions about a few cryptographic assumptions such as decision or computation Diffie Hellman, LRSW and so on.

2.1 Model of traitor tracing schemes

A traitor tracing scheme involves the following entities : the system manager, who is responsible for administrating the system, issuing subscriber's private key, and tracing a pirate decoder, the subscribers of the system and the data suppliers who distribute the encrypted contents to subscribers.

A traitor tracing scheme is comprised of the following procedures.

Join : A procedure that introduces a new user as a subscriber to the system. The join

procedure is a critical component in the context of asymmetric traitor tracing schemes.

Encryption : A procedure that can be used to send encrypted contents to subscribers.

Decryption : A procedure that can be used by any subscriber to decrypt the encrypted contents.

Traitor Tracing : A procedure that can be used by the system manager to reveal the identities of the traitors of a given pirate decoder.

For an asymmetric traitor tracing scheme, one additional party, the judge who verifies that certain subscribers have been implicated in the construction of a pirate decoder, is included. An asymmetric traitor tracing scheme should generate non-repudiated information which can be verified by the judge.

Most of asymmetric traitor tracing schemes use an oblivious polynomial evaluation(OPE) [9,10], where a subscriber can obtain a value of $P(\alpha)$ for a his secret value α and the system manager's secret polynomial $P(x)$. Most of OPE protocols use an oblivious transfer(OT) or its non-interactive version(nOT)[11]. We omit the detailed descriptions on these oblivious protocols.

2.2 Asymmetric Traitor tracing schemes

We briefly describe the scheme of Kiayias and Yung[2] as follows; The second scheme of [7] is based on this scheme, we use it to suggest our scheme.

Join : The system manager chooses a polynomial $f(x,y)=a_0+a_1x+\cdots+a_{2k}x^{2k}+by$ over Z_q. The public key $PK=(g,g^{a_0},g^{-a_1},\cdots,g^{-a_{2k}},g^{-b})$ is published. The system manager and the subscriber u implement OPE protocol over a committed value $< C_u=g^{\alpha_u^C}, sign_u(C_u) >$ of the subscriber. After that, the subscriber obtains a private key $\overrightarrow{K_u}=< f(z_u,\alpha_u), z_u, \alpha_u(=\alpha_u^C+\alpha_u^R) >$ where z_u and α_u^R are randomly chosen by the system manager. The private key $\overrightarrow{K_u}$ is not known (in its entirety) by the system manager, thus asymmetric property can be acquired. Instead, the system manager holds a non-repudiable commitment $< C_u=g^{\alpha_u^C}, sign_u(C_u) >$ of the subscriber to the secret portion of the subscriber's private key.

Encryption : A session key s is encrypted as follows. $C=(s\cdot g^{a_0r}, g^r, g^{-a_1r}, \cdots, g^{-a_{2k}r}, g^{-br})$ where r is a random element of Z_q.

Decryption : Given a ciphertext C, the subscriber u can compute the session key s with his private key as follows.

$$s=\frac{s\cdot g^{a_0r}}{(g^r)^{f(z_u,\alpha_u)}(g^{-a_1r})^{z_u}\cdots(g^{-a_{2k}r})^{z_u^{2k}}(g^{-br})^{\alpha_u}}$$

Traitor tracing : When a pirate decoder containing the pirate key $\overrightarrow{K}=\sum_{i=1}^{t}\mu_i\overrightarrow{k_{u_i}}$ (where $\overrightarrow{k}_{u_1},\cdots,\overrightarrow{k}_{u_t}$ are private keys of the subscribers $u_1,\cdots,u_t$ respectively) is confiscated, The system manager inputs integers $z_1,\cdots,z_n$ to the tracing algorithm. The output is the vector $\overrightarrow{\nu}=<\nu_{u_1},\cdots,\nu_{u_t}>$, where $\nu_{u_i}=\mu_i$ for $i=1,\cdots,t$ and $\nu_i=0$ for all $i\in\{1,\cdots,n\}-\{u_1,\cdots,u_t\}$. Then system manager sends the $\overrightarrow{\nu}, K', \alpha_{u_1}^R,\cdots,\alpha_{u_t}^R$ to judge, where $K'=(\mu_1\alpha_{u_1}+\cdots+\mu_t\alpha_{u_t})$. A judge checks whether $\prod_{i=1}^{t}(C_{u_i}g^{\alpha_{u_i}^R})$

$\stackrel{?}{=} g^K$ for identifying the correctness.

2.3 Bilinear groups

There are many anonymous credential systems and traitor tracing schemes which are defined over the bilinear groups. We briefly review the necessary facts about them.

1. G_1 and G_2 are two (multiplicative) cyclic groups of prime order q.
2. g_1 is a generator of G_1 and g_2 is a generator of G_2,
3. e is a bilinear map $e: G_1 \times G_2 \rightarrow G_T$.

A bilinear map is a map $e: G_1 \times G_2 \rightarrow G_T$ with the following properties:

1. Bilinear: for all $u \in G_1, v \in G_2$ and $a,b \in Z$, $e(u^a, v^b) = e(u,v)^{ab}$.
2. Non-degenerate: $e(g_1, g_2) \neq 1$.

We say that (G_1, G_2) are bilinear groups if there exists a group G_T and a bilinear map $e: G_1 \times G_2 \rightarrow G_T$ as above, and e and the group action in G_1, G_2, and G_T can be computed efficiently.

In the followings, we use a notation of $g := e(g_1, g_2)$, which is a generator of a cyclic group of G_T and use a notation of $BG(q)$ and $G_T(q)$ to denote a bilinear group of order q and a group of order q which is mapped from bilinear groups by a bilinear map.

2.4 Anonymous credential system

There are many cryptographic ways to treat anonymity and some of them are e-cash, group signature, k-times anonymous authentication and anonymous credential system [12,13,8]. Among them, an anonymous credential system treats a certificate as several qualifications which can be used multiple times without linkage between the use, so that it is useful to various applications.

In 2004, Camenisch and Lysyanskaya[8] proposed a series of secure signature schemes over the bilinear groups, which lead to anonymous credential systems. In this paper we use the scheme A, where the security is based on the LRSW assumption. An anonymous credential system using the scheme A can be composed as follows.

Key Generation : An organization generates $PK_{Enc} = (q, G_1, G_2, G_T, g_1, g_2, g, e)$.

It then chooses $x \in Z_q$ and $y \in Z_q$, and sets $SK = (x,y)$, $PK = (X = g_2^x, Y = g_2^y)$.

Issuing a certificate : On the committed input $M = g_1^m$, an organization chooses a random $r \in Z_q$, and outputs a certificate $\sigma = (a,b,c) = (g_1^r, a^y, a^x M^{rxy} = a^{x+xym})$.

A user verifies the validity of the certificate $\sigma = (a,b,c)$ by checking that $e(a, Y) = e(b, g_2)$, and

$e(a,X) \cdot e(b,X)^m = e(c, g_2)$ hold.

Proving ownership : A user chooses $r', r \in Z_q$ at random, generates $\tilde{\sigma} = (\tilde{a}, \tilde{b}, \hat{c}) = (a^{r'}, b^{r'} c^{rr'})$ and sets

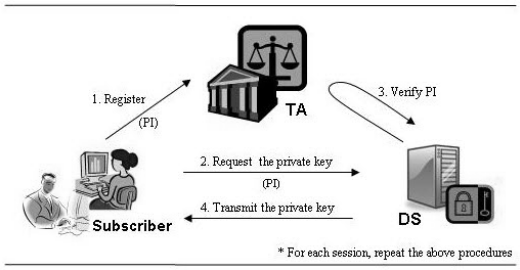
$v_a = e(\tilde{a}, X), v_b = e(\tilde{b}, X), v_c = e(\hat{c}, g_2)$. A user executes a zero-knowledge proof of knowledge of (ω, ρ) for $v_c^{\rho} = v_a \cdot v_b^{\omega}$.

An organization verifies the validity of the certificate $\tilde{\sigma} = (\tilde{a}, \tilde{b}, \hat{c})$ by checking that $e(\tilde{a}, Y) = e(\tilde{b}, g_2)$ and the proof of knowledge.

III. Flaws of the two schemes in [7]

In this Section, we show that the two schemes of [7] have flaws. They introduced

Trust Authority and the pseudo-identity of the subscriber to achieve anonymity. Their resultant schemes are referred to anonymous asymmetric traitor tracing schemes. Let the system manager be SM who is responsible for administrating the system, Trust Authority be TA, the data supplier be DS and the pseudo-identity be PI for short. The Join procedure of the two schemes are depicted in the [figure 1]



[figure1] Join protocol of two schemes in [7]

3.1 The first schemes of [7] is not secure

The first scheme is based on the one-time use scheme of [4], i.e., given a ciphertext, since a subscriber sends all the coefficients of $f(x)$ obviously, therefore DS should change $f(x)$ into another one at the next session and redistribute every subscriber's key. Let us briefly describe the scheme.

Join : A subscriber registers his ID, PI at TA. Then TA publishes PI on the bulletin board and stores ID. DS generates a key generation polynomial

$f(x)=a_0+a_1x+\cdots+a_kx^k$ over Z_q where q is a prime. When a subscriber requests a private key for his PI, DS can verify the PI by accessing the bulletin board of TA. After that, DS selects two random numbers u_0, u_1 and obliviously transfers one of the two pairs of $(u_0,f(u_0))$, $(u_1,f(u_1))$ using a nOT. A subscriber obtains exactly one of the two private keys $(u_0,f(u_0))$, $(u_1,f(u_1))$, while DS does not know which key a subscriber extracts.

Encryption : For a session key s, DS computes the ciphertext $C=(s+a_0, a_1,\cdots,a_k)$.

Decryption : A subscriber can compute $s=(s+a_0+a_1u+\cdots+a_ku^k)-f(u)$.

Traitor Tracing : When a pirate decoder is confiscated, the pirate key $(u_i,f(u_i))$ is exposed. DS searches for PI corresponding to the u_i, and transmits it to TA, then TA sends out the traitor's ID.

Anonymity: Since DS knows only the PI, this scheme achieves the anonymity of the subscriber, and after a session, the subscriber should change his PI to avoid the link.

Flaws: This scheme is vulnerable to the linear attack within a session. The t subscribers $(u_1,\cdots,u_t)$ can collude to make a pirate key $\vec{K}$ with their keys $(u_1,f(u_1)),\cdots,(u_t,f(u_t))$ such that

$$\vec{K}=(\sum_{i=1}^{t}\mu_i,\sum_{i=1}^{t}\mu_iu_i,\cdots,\sum_{i=1}^{t}\mu_iu_i^k,\sum_{i=1}^{t}\mu_if(u_i)),$$

where $\mu_1,\cdots,\mu_t$ are random elements of Z_q, and satisfy $\sum_{i=1}^{t}\mu_i=1$. Given a ciphertext C, the pirate key $\vec{K}$ can be used to compute the session key s as follows:

$$s=(s+a_0+a_1\sum_{i=1}^{t}\mu_iu+\cdots+a_k\sum_{i=1}^{t}\mu_iu_i^k)-\sum_{i=1}^{t}\mu_if(u_i)$$
$$=(s+a_0-(\mu_1+\cdots+\mu_t)a_0)$$

If we raise the degree of $f(x)$ to $2k$, then there exists an efficient algorithm, which is

based on the linear codes decoding, to find the private keys of the traitors from the such a linear combination[2].

3.2 The second scheme of [7] is over-loaded

The second scheme is a public-key scheme. Let us briefly describe the scheme.

Join : A subscriber u registers his ID, PI, and a commitment $C_u = g^{\alpha_u^C}$ u at TA. Then TA publishes PI, a commitment on the bulletin board and stores ID. DS generates a polynomial $f(x,y) = \sum_{i=0}^{2k} a_i x^i + by$ over Z_q, and publishes a public key for encryption

$$PK_{Enc} = (g, g^{a_0}, g^{-a_1}, \cdots, g^{-a_{2k}}, g^{-b}).$$

A subscriber u requests a private key for his PI and DS verifies the PI on the bulletin board at TA. A subscriber obtains a private key $\overrightarrow{K_u} = < f(z_u, \alpha_u), z_u, \alpha_u >$ through OPE, where $\alpha_u = \alpha_u^C + \alpha_u^R$ and z_u, α_u^R are randomly chosen by DS.

Encryption, Decryption, Traitor Tracing : These procedures are equal to those of the scheme of [2] except for the replacement of SM by DS.

Anonymity : A subscriber should change his PI to avoid the link.

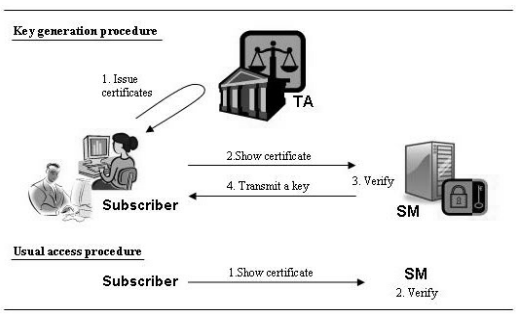
Flaws : In public-key traitor tracing schemes, only SM distributes private keys to the subscribers and publishes a public key in order to enable any DS or subscriber can encrypt data. But in the second scheme, DS plays this role, in this case every DS can publish their public keys for encryption. Therefore the scheme will have inefficient public key size, so that it becomes a non-public-key scheme. This flaw can be corrected easily by replacing DS by SM, then the scheme will satisfy public-key scheme

It is not clearly described whether the key generation polynomial is changed or not in this scheme. But, if one uses a fixed polynomial then a subscriber can obtain several private keys by repeatedly changing his PI and requesting the private keys. Since it leads to a total break of the system under the k colluders, to avoid such operation SM is forced to change the polynomial for each time that a subscriber changes his PI and requests a new private key. This framework restricts an application area and drops the system efficiency.

IV. Constructing an Asymmetric Tratior Tracing Schemes with Anonymity

In this Section we discuss how to construct the concept of anonymity with traitor tracing schemes. We suggest a new framework and then construct a concrete scheme for practical use.

4.1 The new framework



[figure2] The new framework including Issuing certificates, Key generation, Usual access, Encryption/Decryption, Traitor tracing

Abstractly our suggestion is constructing an anonymous credential system and a traitor tracing scheme independently. Through this approach, our suggestion shows that SM and a subscriber can avoid lasting

access to TA. This is an essential requirement of anonymous credential systems, but the schemes of [7] can not satisfy it. Furthermore since we can combine them independently, we can select proper schemes which reflect various requirements of various applications. For example, if the number of use is limited at k times, we can use a k-times anonymous authentication scheme, or if we want to integrate various requirements in a certificate, it will be proper to use an anonymous credential system. Our new framework is depicted in the [figure 2].

We describe our framework as follows.

Issuing certificates : A subscriber sends a commitment with his ID to TA. TA issues certificates bound up with the commitment and sends them to the subscriber.

Key generation : A subscriber shows certificates to SM, then SM checks the validity of the certificates and transmits a private key to the subscriber.

Usual access : A subscriber shows certificates to SM, SM checks the validity and gives permission to access the system.

Encryption and decryption : DS can upload the encrypted contents using a public key of SM, and a subscriber decrypts them using a his private key.

Traitor tracing : Given a pirate decoder, SM traces a traitor in cooperating with TA if needed. As a result, in our framework, we can keep a fixed key generation polynomial and a subscriber obtains a private key from SM only at the key generation procedures. Moreover, Since a subscriber can use certificates many times to avoid the leakage of linking information without contacting with TA, and SM can check the validity of the certificates by itself, TA need not be always online, this relieves the burden of TA.

4.2 A concrete scheme

In this Subsection, the [table 1] shows the characteristics of some parts of our considerations.

[Table1] Characteristics of schemes: $G(q)$ means a group of order q, $BG(q)$ means a bilinear group of order q. We focus on the schemes[2,8].

	scheme	algebraic structure	one-show	multiple-show
Anonymous credential system	[12]	$G(q)$	○	×
	[13]	Z_{pq}	○	○
	[8]	$BG(q)$	△	○

	scheme	algebraic structure	symmetry	resiliency
Traitor tracing scheme	[2]	$G(q)$	asymmetric	k
	[14]	$BG(q)$	asymmetric	k

We pursue the harmony between the schemes rather than just assembling some of them quite independently and we are forced to use a common algebraic structure. For example, the [13] scheme has a good property of supporting both one-show and multiple-show but since it is defined over a different algebraic structure, Z_{pq}, it is difficult to be harmonized with existing traitor tracing schemes. But, in the case of [8], although the method of one-show is inefficient, it becomes harmonized with [2] over G_T and [14], where $G_T(q)$ means an embedding group of order q from a bilinear group.

As a concrete construction, basically, we try to apply two schemes[2, 8]. Since there is only one issuing organization TA in our framework, the scheme A is sufficient for us to use. In [8] they used a zero-knowledge proof of knowledge, which requires an inefficient interactions. To raise the

efficiency we change the proof of knowledge. In the case of the asymmetric traitor tracing scheme [2], we add some process to guarantee that a subscriber uses a same secret both for a commitment and for his private key. From these discussions we suggest our construction as follows.

Setup :

1. TA generates $PK_{BG} = < q, G_1, G_2, g_1, g_2, e >$ for bilinear groups, each of which is defined as in the preliminaries. TA also selects a secret key $x, y \in Z_q$ and sets $X = g_2^x, Y = g_2^y$ and $g = e(g_1, g_2)$. TA publishes the $PK_{BG}, X, Y \in G_2$ and $g \in G_T$.

2. SM selects a random $a_0, a_1, \cdots, a_{2k}, b_1 \in Z_q$ for a collusion limit k.
The SM sets $f(x,y) = P(x) + b_1 y$ and $P(x) = a_0 + a_1 x + \cdots + a_{2k} x^{2k}$ and publishes $PK_{Enc} = < g, g^{a_0}, g^{a_1}, \cdots, g^{a_{2k}}, g^{b_1} >$.

Issuing a certificate :

1. A subscriber selects a secret $w \in Z_q$ and registers himself with his ID and g_1^w.

2. TA selects a random α and sets the certificate $\sigma = (a,b,c) = (g_1^{\alpha}, a^y, a^x (g_1^w)^{\alpha xy})$. TA sends the σ to a subscriber.

3. A subscriber agrees on the certificate if the following equations hold.

$$e(a, Y) = e(b, g_2)$$

$$e(a,X) e(b,X)^w = e(c, g_2)$$

Key generation :

1. A subscriber sets $v_b = e(b,X)$, selects a random t and sets $T = v_b^t$ and sends $T, C_u = g^w$ and σ to SM.

2. SM checks if an equation
$e(a, Y) = e(b, g_2)$
holds. If it holds, SM selects a random β and
sends it to the subscriber.

3. A subscriber replies with $z = t + \beta w$.

4. SM sets $v_a = e(a,X)$, $v_b = e(b,X)$, $v_c = e(c, g_2)$ and checks if an equation
$$(v_a v_c^{-1})^{\beta} \cdot v_b^z = T$$
holds. If it holds SM checks whether the certificate is the first or not by the equation $e(a,c^*) = e(a^*,c)$, where a^* and c^* are parts of the certificate in the current subscriber list. If it is new one, SM accepts the validity and selects a random z_u and generates a private key $< f(z_u, \alpha_u), z_u, \alpha_u >$ with the subscriber by OPE, where $\alpha_u = w + \beta$. Note that SM doesn't know α_u nor $f(z_u, \alpha_u)$. SM updates the subscriber list with these whole values.

5. A subscriber computes $g^{f(z_u, \alpha_u)}$ and sends it to SM.

6. SM checks if a subscriber uses the same w for both C_u and OPE by the equation $g^{f(z_u, \alpha_u)} = g^{P(z_u)} \cdot (C_u \cdot g^{\beta})^{b_1}$.

Usual access :

1. A subscriber selects $r', r, t_1, t_2 \in Z_q$ and sets the following values.
$\tilde{\sigma} = (\tilde{a}, \tilde{b}, \tilde{c}) = (a^{r'}, b^{r'}, c^{rr'})$,
$v_a = e(\tilde{a}, X), v_b = e(\tilde{b}, X)$
$T = v_a^{t_1} v_b^{t_2}$
A subscriber sends $\tilde{\sigma}$ and T to SM.

2. SM checks if an equation $e(\tilde{a},Y)=e(\tilde{b},g_2)$ holds. If it holds, SM selects a random γ and sends it to the subscriber.

3. A subscriber replies with $z_1=t_1+\gamma r,\quad z_2=t_2+\gamma r w.$

4. SM sets $v_a=e(\tilde{a},X)$, $v_b=e(\tilde{b},X)$, $v_c=e(\hat{c},g_2)$ and checks if an equation $v_a^{z_1}v_b^{z_2}v_c^{-\gamma}=T$ holds. If it holds SM accepts the validity and permits a subscriber access to the encrypted contents.

Encryption and decryption :

1. DS selects a random r and a session key s. Given PK_{Enc}, DS broadcasts the ciphertext $<h,h_0,\cdots,h_{2k},h'>=<g^r,s\cdot g^{a_0 r},g^{a_1 r},\cdots,g^{a_{2k} r},g^{b_1 r}>$

2. A subscriber computes s from the following equation. $s=(\prod_{i=1}^{2k} h_i^{z_u{}^i})\cdot h'^{\alpha_u}\cdot h^{-f(z_u,\alpha_u)}$

Traitor tracing :

1. SM searches for $z_u{}'$s of all traitor through the tracing algorithm in [2] and sends them to TA.

2. TA finds the corresponding identities.

4.3 Security proofs

In the above scheme, since we remove the zero-knowledge property in the process of proving knowledge, we have to check if this approach brings into some weakness or not. In this Subsection we prove that our protocols for proof of knowledge are secure on the impersonation attack and unlinkable.

To prove the security on the impersonation attack, we take two steps. At first we prove that our protocols are really proof of knowledge, and then we reduce the impersonation to the discrete logarithm problem.

Theorem 1. *Our protocols for proving certificates are proof of knowledge.*

Proof. It is sufficient to show that we can extract the witnesses. In the first protocol of the key generation process, for the same σ, same T, different challenges β,β' and different responses z,z', we can extract w by the equation of $w=\frac{z-z'}{\beta-\beta'}$ □

In the second protocol of usual access, similarly for the same σ, same T, different challenges γ,γ' and different responses z_1,z_2,z'_1,z'_2, we can extract w and r by the equations of

$$r=\frac{z_1-z'_1}{\gamma-\gamma'},\quad w=\frac{z-z'}{\beta-\beta'}.$$

Theorem 2. *For the above protocols, if the impersonation attack succeeds with non-negligible probability, then we can solve a discrete logarithm problem(DLP) with non-negligible probability.*

Proof. Since the protocols have same structure, we only state for the second protocol. Let A be an adversary who tries to impersonate and B be an adversary who tries to solve the discrete logarithm $u=\log_{g_1} U$ for a given U. B randomly selects $\alpha,x,y\in Z_q$ and composes a certificate like following.

$$\tilde{\sigma}=(\tilde{a},\tilde{b},\tilde{c})=(g_1^{\alpha},\tilde{a}^{y},U)$$

If A succeed one time then he can also

succeed two times, which we can know from the arguments of [9]. Since this protocol is a proof of knowledge, then A can extract the witnesses w and r. Now, using them, B can compute the following equation.

$$U=\hat{c}=\tilde{a}^{rx(1+yw)}=g_1^{\alpha rx(1+yw)}$$
$$\therefore u=\alpha rx(1+yw) \quad \square$$

We show that the proving protocol for usual access guarantees the unlinkability of a certificate. For any two transits of the protocol, no one can decide whether the owners are the same or not. To enhance the privacy, we require TA cannot link the certificates too.

Theorem 3. *The proving protocol of the usual access is unlinkable.*

Proof. We discuss the probability ensemble. Let's consider two transits of the different subscribers who use r,w and r^*,w^* as their witnesses, respectively. Note that the unlinkability is originated from the difference w and w^*. We will convert the value of w^* into w and investigate the changes of other values, and deduce that we cannot distinguish two probability spaces. For the following two transits

$$<\sigma,\pi> \;=\; <(\tilde{a},\tilde{b},\tilde{c}),(T,\gamma,z_1,z_2)>,$$
$$<\sigma^*,\pi^*>=<(\tilde{a}^*,\tilde{b}^*,\tilde{c}^*),(T^*,\gamma^*,z_1^*,z_2^*>,)$$

If there are relations of $\tilde{a}=g_1^{\alpha},\tilde{a}^*=g_1^{\alpha^*}$, and for the $\tilde{r}$ such that

$\tilde{r}(1+wy)=r^*(1+w^*y)$, we can write the σ and σ^* as

$$\sigma=(g_1^{\alpha},Y^{\alpha},X^{\alpha r(1+wy)})$$
$$\sigma^*=(g_1^{\alpha^*},Y^{\alpha^*},X^{\alpha^* r^*(1+w^*y)})=(g_1^{\alpha^*},Y^{\alpha^*},X^{\alpha^*\bar{r}(1+wy)})$$

It is clear that the two ensembles of (α,α^*,r,r^*) and $(\alpha,\alpha^*,r,\bar{r})$ are same. This means that it is unable to decide the linkability only by σ and σ^*. Now, let $z_1^*=t_1^*+\gamma^* r^*$ and $z_2^*=t_2^*+\gamma^* r^* w^*$. For these z_1^*,z_2^*, we consider $\overline{z_1}=t_1^*+\gamma^*\bar{r}$ and $\overline{z_2}=t_2^*+\gamma^*\bar{r}w$. Then we can also know the two ensembles of $(T^*,\gamma^*,z_1^*,z_2^*)$ and $(T^*,\gamma^*,\overline{z_1},\overline{z_2})$ are same. This completes the proof. $\square$

4.4 Applications

Our framework of constructing asymmetric traitor tracing schemes with anonymity will be useful in the following circumstances; (1) There is an enormous amount of data. (2) Qualified subscribers can access this data. (3) To protect subscribers' privacy and to detect some traitors.

As the first application, we consider an online digital contents service system, which is a software-based system. In this system, various DS furnish digital contents like newspapers, music, pictures, movies, etc. through encryptions by SM's public key. When a subscriber registers TA, he obtains a certificate according to his credentials like age, gender, the term of validity, the number of use, etc. Each subscriber has access to the system and SM generates a private key and executes access control by verifying whether his certificate includes the qualification. If some subscribers collude to make a pirate key, then one of them can be traced by the cooperation of SM and TA. Nowadays there are many sites which support these kinds of applications, but most of them don't consider the privacy nor traitor tracing.

As the second application, we consider a video on demand or a pay-TV system with differed payment, on a hardware-based system. TA uses a credential system which supports traceability of a subscriber only by TA. There are various DS such as sports

channel, movie channel and music chann

els, etc. They can encrypt contents using the SM′s public key and broadcast encrypted contents in the system. A subscriber obtains his certificate from TA beforehand and he watches the video or TV. Since SM cannot know who watches what channels, subscriber′s privacy is protected. After that, SM can delegate TA to ask for payment of the subscriber

V. Conclusion

In this paper, to enhance the subscribers′ privacy further, we considered how to introduce anonymity to traitor tracing schemes. At first, we pointed out the schemes of [7] have flaws and argued where these flaws stem from. Although the schemes of [7] were behind the times and insignificant in this field, the concept of asymmetric traitor tracing scheme with anonymity would be meaningful. In general, the property of anonymity might be conceptually conflicting with traceability of the traitors, hence it was not easy to design an anonymous traitor tracing scheme. As a result, we construct a new concept of annoymous asymmetric scheme, in which subscribers can hide their identities from the system manager by using an anonymous credential system and an asymmetric traitor tracing scheme. To raise the efficiency, we modify the schemes and prove the security, furthermore we show two practical application. Therefore our scheme can be an option to application where enhancing the subscribers′ privacy.

참고문헌

[1] B. Pfitzmann, "Trials of traced traitors," Information Hiding′96, LNCS 1174, pp 49-64, June. 1996.

[2] A. Kiayias and M. Yung, "Breaking and repairing asymmetric public key traitor tracing," 2002 ACM Workshop on Digital Rights Management, LNCS 2696, pp. 32-50, Nov. 2003.

[3] H. Komaki, Y. Watanabe, G. Hanaoka and H. Imai, "Efficient asymmetric self-enforcement scheme with public traceability," PKC 2001, LNCS 1992, pp. 225-239, Feb. 2001.

[4] K. Kurosawa and Y. Desmedt, "Optimum traitor tracing and asymmetric scheme," EUROCRYPT′98, LNCS 1403, pp. 145-157, June. 1998.

[5] Y. Watanabe, G. Hanaoka and H. Imai, "Efficient asymmetric public key traitor tracing without trusted agents," CT-RSA 2001, LNCS 2020, pp. 392-407, April. 2001.

[6] E. Magkos, P. Kotzanikolaou and V. Chrissikopoulod, "An asymmetric traceability scheme for copyright protection without trust assumptions," Electronic Commerce and Web Technologies, LNCS 2115, pp. 186-195, Sep. 2001.

[7] E. Y. Choi, J. Y. Hwang and D. H. Lee, "An anonynous asymmetric public key traitor tracing scheme," E-Commerce and Web Technologies, LNCS 2738, pp. 104-114, Sep. 2003.

[8] J. Camenisch and A. Lysyanskaya, "Signature schemes and anonymous credentials from bilinear maps," CRYPTO 2004, LNCS 3152, pp. 56-72, Aug. 2004.

[9] Y. C. Chang and C. J. Lu, "Oblivious polynomial evaluation and oblivious neural learning," ASIACRPT′99, LNCS 2248, pp. 369-384, Dec. 2001.

[10] M. Naor and B. Pinkas, "Oblivious transfer and polynomial evaluation," the 31th ACM Symposium on the Teory of Computing, In: Proc. ACM Symposium on Theory of Computing, pp. 245-254, 1999.

〔11〕 M. Bellare and S. Micali, "Non-Interactive oblivious transfer and applications," CRYPTO'89, LNCS 435, pp. 544-557, Aug. 1989.
〔12〕 S. Brands, "A technical overview of digital credentials," http://www.credentica.com/technology.html, 2002.
〔13〕 J. Camenisch and A. Lysyanskaya, "An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation," EUROCRYPT 2001, LNCS 2045, pp. 93-118, May. 2001.
〔14〕 V. T$\hat{O}$, R Safavi-Naini and F.Zhang, "New traitor tracing schemes using bilinear map," 2003 ACM Workshop On Digital Rights Management, ISBN 1-58113-786-9, pp. 67-76, Oct. 2003.
〔15〕 A. Fiat and A. Shamir, "How to prove yourself: practical solutions to identifcation and signature problems," CRYPTO 1998, LNCS 263, pp. 186-194, Aug. 1986.
〔16〕 D. Boneh, X. Boyen, "Short signatures without random oracles," EUROCRYPT 2004, LNCS 3027, pp. 56-73, May. 2004.

〈著者紹介〉

이 문 식 (Moonsik Lee) 정회원
2001년 2월: 서울대학교 수리과학부 학부
2004년 2월: 서울대학교 수리과학부 석사
2010년 2월: 서울대학교 수리과학부 박사
2010년 2월 ~현재: 공군사관학교 기초과학과 수학교수
〈관심분야〉 정보보호, 암호학

강 순 부 (SunBu Kang) 정회원
1996년 2월: 서울대학교 수리과학부 학부
1999년 2월: 포항공대 수학과 석사
2004년 8월: 포항공대 수학과 박사
2004년 8월 ~현재: 공군사관학교 기초과학과 수학교수
〈관심분야〉 수치해석, 정보보호, 암호학

이 주 희 (Ju-Hee Lee) 정회원
1996년 2월: 한남대학교 수학과 학사
2002년 2월: 이화여자대학교 수학과 석사
2010년 8월: 이화여자대학교 수학과 박사
2010년 9월 ~2012년 5월: 이화여자대학교 수리과학연구소 연구원
2012년 6월 ~현재: 국가수리과학연구소 연구원
〈관심분야〉 정보보호, 암호론