A Traitor Tracing Scheme Based on the RSA System

Bo Yang[1], Hua Ma[2], and Shenglin Zhu[1]
(Corresponding author: Bo Yang)

College of Information, South China Agricultural University [1]
Guangzhou 510640, P. R. China (Email: byang@scau.edu.cn)
School of Science, Xidian University [2]
Xian, Shaanxi Province, 710071, P. R. China (E-mail: hma@mail.xidian.edu.cn)

(Received Dec. 20, 2005; revised and accepted Jan. 27, 2006)

Abstract

Traitor tracing schemes constitute a very useful tool against piracy in the context of digital content broadcast. In such multi-recipient encryption schemes, the data-suppliers can reveal the identities of the subscribers that were implicated in the construction of a pirate-device that illegally receives the digital content. In this paper, a traitor tracing scheme based on the RSA system is proposed. The scheme does not rely on any trusted third party, and is k-collusion resistant, can provide black-box confirmability, and revoke any traitor's personal decryption key (without the limitation to threshold) without updating the personal decryption key of the remaining subscribers. And the scheme is more efficient than the ElGamal-like scheme according to complexity overload. Its security is the same as that of the RSA system.

Keywords: Broadcast, encryption, RSA, traitor tracing

1 Introduction

In an open broadcast network, a data supplier transmits digital contents to a large number of users in such a way that only authorized users can extract the contents. To prevent unauthorized users from accessing data, the data supplier will encrypt data and provide only the authorized users with decryption key. However, some unauthorized subscribers *(pirates)* may obtain some decryption keys from one or more authorized subscribers *(traitors)*. The goal of a traitor tracing scheme is to provide a method so that the data-suppliers, given a pirate decoder, are able to recover the identity of the traitors [1, 2, 3, 4, 5, 6, 7, 9, 10].

A traitor tracing scheme is "k-collusion resistant", which satisfies that no coalition of at most k users can create a pirate decoder, such that none of the traitors will be detected, i.e., at least one of the traitors can always be identified.

In [6], an ElGamal-like traitor tracing scheme is given, many other traitor tracing schemes [1, 2, 3, 4, 5, 7, 9, 10] are constructed with their structure based on the one of [6]. In this paper, as one method of approach, based on the RSA system we present an efficient traitor tracing scheme without the need of any trusted third party. The scheme is k-collusion resistant, can revoke any traitor's personal decryption key (without the limitation to threshold) without updating the personal decryption key of the remaining subscribers. The scheme is as secure as the RSA system.

In Section 2, we describe the ElGamal-like traitor tracing scheme and its flaws. Our construction is described in Section 3. We analyze the security and the efficiency of our proposal in Section 4. Finally, conclusions are given in Section 5.

2 The ElGamal-like Traitor Tracing [6]

Let p be a prime power, q be a prime such that $q \mid p-1$, $q > n$, (where n is the number of users) and g be a qth root of unity over $GF(p)$.

2.1 Initialization

DS (Data Supplier) chooses a random polynomial $f(x) = a_0 + a_1 x + \cdots + a_k x^k$ over Zq and computes $y_0 = g^{a_0}, y_1 = g^{a_1}, \cdots, y_k = g^{a_k}$.

Let $e_T = (p, g, y_0, y_1, \cdots, y_k)$ be a public key.

DS gives $f(i)$ to authorized user u_i as personal decryption key $e_i, 1 \le i \le n$.

2.2 Distributing a Session Key

For a session key s, DS computes a header as $h(s,r) = (g^r, sy_0^r, y_1^r, \cdots, y_k^r)$ where r is a random number. Then DS broadcasts $h(s,r)$. Each user u_i computes s from

$h(s,r)$ and e_i as follows:

$$\begin{aligned}\{sy_0^r \times (y_1^r)^i \times \cdots \times (y_k^r)^{i^k}\}/(g^r)^{f(i)} &= s(g^r)^{f(i)}/(g^r)^{f(i)} \\ &= s.\end{aligned}$$

2.3 Broadcasting Encrypted Data

To send actual plaintext data M, DS broadcasts $C = E_s(M)$, where E is a symmetric key encryption function. Every authorized user can recover s, and then decrypts C to obtain M.

2.4 Detection of Traitor

When a pirate decoder is confiscated, the pirate key e_p is exposed. If e_p contains $\langle j, f(j)\rangle$, then DS decides that user u_j is a traitor.

2.5 Flaws in the ElGamal-like Traitor Tracing [1]

The scheme cannot resist the attack, named convex combination attack, described below.

Let $f(i_1), f(i_2), \cdots, f(i_k)$ be k personal decryption key, $w = [u, w_0, w_1, \cdots, w_k]$ be any convex combination of the vectors $v_1, \cdots, v_k$, defined by:

$$\begin{aligned} v_1 &= [f(i_1), 1, i_1, i_1^2, \cdots, i_1^k] \\ \vdots & \qquad\qquad \vdots \\ v_k &= [f(i_k), 1, i_k, i_k^2, \cdots, i_k^k]. \end{aligned}$$

That is $w = \sum_{i=1}^k a_i v_i$, where $\sum_{i=1}^k a_i = 1$. The w is not a legitimate personal decryption key, but it can be used to decrypt any ciphertext $c = [a, b_0, b_1, \cdots, b_k]$ since $b_0^{w_0} \cdots b_k^{w_k}/a^u = s$.

By increasing the degree of f from k to $2k-1$, k-collusion resistance can be achieved [5, 9].

3 RSA-like Traitor Tracing

3.1 RSA System [8]

Let p and q be two secret large primes, $N = p \times q, \varphi(N) = (p-1)\times(q-1)$, e be an integer selected randomly, satisfied $\gcd(e, \varphi(N)) = 1$ and $0 < e < \varphi(N)$, d be another integer satisfied $de \equiv 1(\mathrm{mod}\varphi(N))$. Then (e, N) are public key, d is the secret key.

To encrypt a message M, the sender computes $C = M^e(\mathrm{mod}N)$. To decrypt the message C, the receiver computes $M = C^d(\mathrm{mod}N)$.

3.2 Initialization

DS chooses an RSA system, in which the public key is (e, N), secret key is d. Then DS chooses a secret polynomial $f(x) = \sum_{i=0}^k a_i x^i(\mathrm{mod}\varphi(N))$, satisfied $f(0) = d$. DS gives $f(i)$ to authorized user u_i as personal decryption key e_i.

It is easy to obtain asymmetry by using OPE (oblivious polynomial evaluation), in which a authorized user u_i randomly chooses an non-zero integer α_i and computes his personal key $f(\alpha_i)$ using OPE in such a way that the DS does not learn α_i and the user u_i does not gain any additional information on $f(x)$. The method is similar to the one of [5] or the one of [10].

3.3 Distributing a Session Key

Let Φ be a set of authorized users, I be a set of integers. For a session key $s(0 < s < N)$, DS computes a header as

$$\begin{aligned} h(s,N) =< & s^e \bmod N, (x_1, (s^e)^{f(x_1)} \bmod N), \\ & (x_2, (s^e)^{f(x_2)} \bmod N), \cdots, (x_k, (s^e)^{f(x_k)} \bmod N) > \end{aligned}$$

where $x_1, x_2, \cdots, x_k$ are randomly chosen from $I - \phi$ and different in pairs. Then DS broadcasts $h(s, N)$. Each user u_i computes s from $h(s, N)$ and e_i as follows.

Let $x_{k+1} = i$, then $f(x_{k+1}) = f(i)$. Compute $\Pi_{t=1}^{k+1}[(s^e)^{f(x_i)}]^{\lambda_t} \bmod N = s$, where

$$\lambda_t = \prod_{j=1 \atop j\neq t}^{k+1} \frac{x_j}{x_j - x_i}.$$

This is because

$$\prod_{t=1}^{k+1}[(s^e)^{f(x_t)}]^{\lambda_t} \bmod N = (s^e)^{\sum_{t=1}^{k+1} f(x_t)\lambda_t}.$$

From Lagrange interpolation formula, $\sum_{t=1}^{k+1} f(x_t)\lambda_t = f(0) = d \bmod \varphi(N)$, so $\sum_{t=1}^{k+1} f(x_t)\lambda_t = l\varphi(N) + d$, where l is an integer.

$$\begin{aligned} \prod_{t=1}^{k+1}[(s^e)^{f(x_t)}]^{\lambda_t} \bmod N &= (s^e)^{\sum_{t=1}^{k+1} f(x_t)\lambda_t} \\ &= (s^e)^{l\varphi(N)+d} \\ &= (s^{l\varphi(N)+d})^e. \end{aligned}$$

In following, the method to prove $(s^{l\varphi(N)+d})^e = s \bmod N$ is similar to the one in [8]. If s and N are relatively prime, by virtue of Euler's theorem:

$$\begin{aligned} s^{\varphi(N)} &\equiv 1 \bmod N, s^{l\varphi(N)} \equiv 1 \bmod N, \\ s^{l\varphi(N)+d} &\equiv s^d \bmod N. \end{aligned}$$

Suppose $\gcd(s, N) \neq 1$. What does this mean? Because $N = p \times q$, the equality $\gcd(s, N) = 1$ is equivalent to the logical expression (s is not a multiple of p) AND (s is not a multiple of q). Therefore, the expression $\gcd(s, N) \neq 1$ is equivalent to the logical expression (s is a multiple of p) OR (s is a multiple of q). Let $s = cp$ holds for some positive integer c. In this case, we must have $\gcd(s, q) = 1$. Otherwise, we have s a multiple of q, so a multiple of pq, this is a contradiction with $s < N = pq$.

From $\gcd(s,q) = 1$ and Euler's theorem, we have $s^{\varphi(q)} \equiv 1 \bmod q$, so $s^{l\varphi(q)} \equiv 1 \bmod q$, $[s^{l\varphi(q)}]^{\varphi(p)} \equiv 1 \bmod q$, $s^{l\varphi(N)} \equiv 1 \bmod q$. Therefore, there is some integer r such that $s^{l\varphi(N)} \equiv 1 + rq$. Multiplying each side by $s = cp$, $s^{l\varphi(N)+1} = s + rcpq = s + rcN$, $s^{l\varphi(N)+1} \equiv s \bmod N$, $s^{l\varphi(N)+d} \equiv s^d \bmod N$. So

$$\prod_{t=1}^{k+1}[(s^e)^{f(x_t)}]^{\lambda_t} = (s^{l\varphi(N)+d})^e = s^{de} = s \bmod N.$$

The broadcast of encrypted data is the same as the ones of the ElGamal-like traitor tracing.

3.4 Detection of Traitors

3.4.1 Non-black-box Tracing

It is the same as the ones of the ElGamal-like traitor tracing.

In our scheme, no coalition of at most k traitors can generate another personal key from their personal keys and the public information (see Theorem described in below). On the other hand, it seems not to be applicable for a convex combination attack to our scheme, since a session key can be computed by combining $k+1$ shares using the Lagrange interpolation, and simple convex combination of the personal keys of k traitors does not lead to the pirate key.

3.4.2 Black-box Algorithm

Our black-box algorithm does not require any trapdoors of the discrete logarithm, as well as that of [10].

In a trail the DS can convince any arbiter that a pirate decoder contains one of the traitor's personal keys simply by observing its behavior on a new header constituted by the DS (i.e., using the pirate decoder as an oracle).

Let $\Psi = \{j_1, \cdots, j_l\}(l < k)$ be a set of suspected traitors. The DS computes a head $h'(s,N)$ as follows. DS selects randomly $k-l$ pairs $(j_{l+1}, d_1), (j_{l+2}, d_2), \cdots, (j_k, d_{k-l})$ such that none of them is on $f(x)$, then there exists a unique k-degree polynomial $p(x) = \sum_{i=0}^{k} b_i x^i (\bmod \varphi(N))$ passing points:

$$(0,d), (j_1, f(j_1)), \cdots, (j_l, f(j_l)), \\ (j_{l+1}, d_1), (j_{l+2}, d_2), \cdots, (j_k, d_{k-l}).$$

The DS can determine $b_0, b_1, \cdots, b_k$ uniquely by solving the system of $k+1$ equations. Then, the DS computes $h'(s,N)$ as

$$< s^e \bmod N, (x'_1, (s^e)^{p(x'_1)} \bmod N), (x'_2, (s^e)^{p(x'_2)} \bmod N), \\ \cdots, (x'_k, (s^e)^{p(x'_k)} \bmod N) >,$$

where $x'_1, x'_2, \cdots, x'_k$ are randomly chosen from $I - \{0, j_1, \cdots, j_k\}$, different in pairs.

Then the DS observes the output given from the confiscated decoding-box with the input $h'(s,N)$, if the decoding-box possesses a personal key $f(j_i)(1 \le i \le l)$ belonging to the user in Ψ, because $f(j_i)$ is on $p(x)$, the decoding-box can obtain and output s by Lagrange interpolation formula with $p(x)$. Therefore, the arbiter can confirm that the pirate decoder contains the personal key of the accused traitors in a black box fashion by running the above confirmation algorithm on all candidate coalitions among $l(l < k)$ accused traitors.

3.5 Revocation of Traitors

After a pirate decoder is confiscated and the traitors are revealed, we would like to revoke personal decryption key of the traitors.

Assume that $\Lambda_r = \{i_1, \cdots, i_r\}$ is the set of found traitors or revoked subscribers, in which $r = mk + t(0 \le t < k)$. We can revoke their decryption keys without updating the personal decryption key of the remaining subscribers. DS decomposes s into $m+1$ multiplied integers, that is, $s = s_1 \cdots s_m s_{m+1}$, performs the loop as follows.

For $j = 1$ to m do

Computes and broadcasts a header as

$$h(s_j, N) =< s_j^e \bmod N, (i_{(j-1)k+1}, (s_j^e)^{f(i_{(j-1)k+1})} \bmod N), \\ \cdots, (i_{(j-1)k+k}, (s_j^e)^{f(i_{(j-1)k+k})} \bmod N) > .$$

The user $i \in \{i_{(j-1)k+1}, \cdots, i_{(j-1)k+k}\}$ can not obtain s_j.

After the for-loop ends, DS computes and broadcasts another header as

$$\begin{aligned} h(s_{m+1}, N) \quad = \quad & < s_{m+1}^e \bmod N, \\ & (i_{mk+1}, (s_{m+1}^e)^{f(i_{mk+1})} \bmod N), \\ & \cdots, \\ & (i_{mk+t}, (s_{m+1}^e)^{f(i_{mk+t})} \bmod N), \\ & (j_1, (s_{m+1}^e)^{f(j_1)} \bmod N), \\ & \cdots, \\ & (j_{k-t}, (s_{m+1}^e)^{f(j_{k-t})} \bmod N) >, \end{aligned}$$

where $j_1, \cdots, j_{k-t}$ are randomly chosen from $I - \Phi - \Lambda_r$ and different in pairs.

Every user $i \in \Lambda_r$ cannot obtain s because he cannot obtain $s_{\lfloor \frac{i}{k} \rfloor + 1}(i \le mk)$ or $s_{m+1}(i > mk)$, but other users can compute $s_i(i = 1, \cdots, m+1)$ and obtain $s = s_1, \cdots s_m s_{m+1}$ without updating one own personal key.

4 Security and Efficiency

Theorem 1. *The computational complexity for k traitors $\Omega = \{i_1, i_2, \cdots, i_k\}$ of finding a pirate key $f(z)$ such that $z \notin \Omega$, when given the public key $\{e, N\}$, the head $h(s,N)$ and their personal keys $f(i_l)(l = 1, \cdots, k)$, is as hard as to break the RSA system.*

Proof. Let M_1 be the polynomial time algorithm that k traitors would use to find a pirate key $f(z)$ such that

Table 1: A comparison of the times of modular exponentiations in the ElGamal-like scheme and our scheme

	ElGamal-like scheme (data-supplier, user)	Our scheme (data-supplier, user)
Initialization	(k+1,-)	(-,-)
Distributing a session key	(k+2,k+1)	(k+1,k+1)
Black-box algorithm	($2k$,-)	(k+1,-)

$z \notin \Omega$, M_2 be the polynomial time algorithm to breaking the RSA system. At first, it is clear that existence of M_2 implies the existence of M_1. Secondly, suppose that there exists M_1, we will show M_2 by using M_1 as a subroutine. Let the input to M_2 be $\{e, N\}$ and $s^e \bmod N$, M_2 first chooses $d_1, d_2, \cdots, d_k$ at random. Then there exists a unique polynomial $f(x) = \sum_{i=0}^{k} a_i x^i (\mathrm{mod}\varphi(N))$ such that $f(0) = d, f(i_j) = d_j$ for $1 \le j \le k$. M_2 constitutes $h(s, N)$ as

$$\langle s^e \bmod N, (i_1, (s^e)^{d_1} \bmod N), (i_2, (s^e)^{d_2} \bmod N), \cdots, (i_k, (s^e)^{d_k} \bmod N)\rangle.$$

(Note that d anf $f(x)$ are unknown for M_2). M_2 feeds $\{e, N\}$, $s^e \bmod N$ and $h(s, N)$ to M_1. Finally, if M_1 outputs $(z, f(z))$ such that $z \notin \{i_1, i_2, \cdots, i_k\}$, then M_2 can compute s by using Lagrange interpolation with $f(i_{k+1}) = f(z)$, that is, the RSA system is broken. This happens with nonnegligible probability. Thus, the existence of M_1 implies the existence of M_2.

The main complexity overload of our scheme and the ElGamal-like scheme is modular exponentiations. The times of modular exponentiations in two schemes are given in Table 1.

In each pair, the two numbers denote the times of modular exponentiations needed by the DS and user respectively, "-" denotes that the modular exponentiation is not needed. It shows that our scheme is more efficient than the ElGamal-like scheme according to complexity overload.

Table 2 shows a comparison about the decryption key size and the data redundancy size among related works and our proposal. $1/\rho$ and $1/\rho_B$ are defined by

1) $1/\rho \triangleq max\{log|U_i|/log|S| \ : i \in \Phi\}$

2) $1/\rho_B \triangleq max\{log|B|/log|S|\}$,

where U_i denotes the set of all possible subsets of decryption keys, B denotes the set of all possible subsets of the data redundancy, S denotes the set of all possible subsets of the session keys and Φ denotes the set of subscribers of the system. Thus $1/\rho$ is a parameter on the size of each subscriber's decryption key and $1/\rho_B$ is a parameter on the size of the data redundancy. From a brief view of Table 2, it is obvious that our scheme is one of the most efficient schemes. □

Table 2: A comparison of the decryption key size and the data Redundancy size

	$1/\rho$	$1/\rho_B$
[2]	1	$2k$+1
[5]	2	$2k$+1
[6]	1	$2k$+1
[9]	1	$2k$+1
[10]	2	$3k$+1
Proposal	1	k+1

5 Conclusion

In this paper, a traitor tracing scheme based on the RSA system is proposed, it has been shown that the scheme is k-collusion resistant, does not require the involvement of the third trusted party(s), provides black-box confirmability, can revoke any traitor's personal decryption key(without the limitation to threshold) without updating the personal decryption key of the remaining subscribers. And the scheme is more efficient than the ElGamal-like scheme according to complexity overload. The security of our scheme is equivalent to that of the RSA system.

Acknowlegements

This work is supported by the National Natural Science Foundation of China under Grant Nos. 60372046, 60573043 and the Foundation of National Laboratory for Modern Communications under Grant No. 9140c1108010606.

References

[1] D. Boneh and M. Franklin, "An efficient public key traitor tracing scheme," in *Crypto'99*, LNCS 1666, pp. 338-353, Springer-Verlage, 1999.

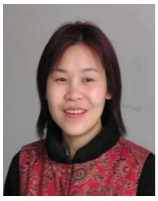
[2] B. Chor, A. Fiat, and M. Naor, "Tracing traitors," in *Crypto'94*, LNCS 839, pp. 257-270, Springer-Verlag, 1994.

[3] A. Kiayias and M. Yung, "Breaking and repairing asymmetric public-key traitor tracing," in *Digital Right Management (DRM'02)*, LNCS 2696, pp. 32-50, Springer-Verlage, 2003.

[4] H. J. Kim, D. H. Lee, and M. Yung, "Privacy against piracy: Protecting two-level revocable P-K traitor tracing," in *ACISP'02*, LNCS 2384, pp. 482-496, Springer-Verlage, 2002.

[5] H. Komaki, Y. Watanabe, G. Hanaoka, and H. Imai, "Efficient asymmetric self-enforcement scheme with public traceability," in *PKC'01*, LNCS 1992, pp. 225-239, Springer Verlag, 2001.
[6] K. Kurosawa and Y. Desmedt, "Optimum traitor tracing and asymmetric schemes," in *Eurocrypt'98*, LNCS 1498, pp. 145-157, Springer-Verlag, 1998.
[7] M. Naor and B. Pinkas, "Efficient trace and revoke schemes," *FC'00*, LNCS 1962, pp. 1-20, Springer-Verlage, 2001.
[8] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 2nd ed., Prentice Hall, Inc. 1999.
[9] W. G. Tzeng and Z. J.Tzeng, "A public-Key traitor tracing scheme with revocation using dynamic shares," in *PKC'01*, LNCS 1992, pp. 207-224, Springer-Verlag, 2001.
[10] Y. Watanable, G. Hanaoka and H. Imai, "Efficient asymmetric public-key traitor tracing without trusted agents," in *Cryptology*, LNCS 2020, pp. 392-407, Springer-Verlag, 2001.

Bo Yang received the B. S. degree from Peking University in 1986, and the M. S. and Ph. D. degrees from Xidian University in 1993 and 1999, respectively. From July 1986 to July 2005 he had been at Xidian University, from 2002, he had been a professor of National Key Lab. of ISN in Xidian University, supervisor of Ph.D. In May 2005, he has served as a Program Chair for the fourth China Conference on Information and Communications Security (CCICS2005). He is currently a professor and supervisor of Ph.D. at College of Information, South China Agricultural University. He is a senior member of Chinese Institute of Electronics (CIE), a member of specialist group on information security in Ministry of Information Industry of P.R.China and a member of specialist group on computer network and information security in Shaanxi Province. His research interests include information theory and cryptography.



Hua Ma received the B.S. and M.S. degree from Xidian University in 1985 and 1990, respectively. Now she is a professor at Xidian university mainly interested in cryptology and network security.

Shenglin Zhu received the B.S. degree from Beijing Normal University in 1993 and the M.S. degree from South China Agriculture University in 2004. Now he is an associate professor at South China Agriculture University and mainly interested in information security and computer application.