Optimum Traitor Tracing and Asymmetric Schemes*

Kaoru Kurosawa[1] and Yvo Desmedt[2]**

[1] Dept. of Electrical and Electronic Eng., Tokyo Institute of Technology
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan
kurosawa@ss.titech.ac.jp, http://tsk-www.ss.titech.ac.jp/~kurosawa/

[2] EE & CS, and the Center of Cryptography, Computer and Network Security
University of Wisconsin - Milwaukee, PO Box 784, WI 53201, USA, and
Dept. of Mathematics, Royal Holloway, University of London, UK
desmedt@uwm.edu, http://www.uwm.edu/~desmedt/

Abstract. A traceability scheme is a broadcast encryption scheme such that a data supplier T can trace malicious authorized users (traitors) who gave a decryption key to an unauthorized user (pirate). This paper first derives lower bounds on the sizes of keys and ciphertexts. These bounds are all tight because an optimum one-time use scheme is also presented. We then propose a multiple-use scheme which approximately meets our bounds. This scheme is proven to be secure as well as much more efficient than the schemes by Chor, Fiat and Naor. Finally, practical types of asymmetric schemes with arbiter are discussed in which T cannot frame any authorized user as a traitor.

1 Introduction

In such applications, as pay TV, CD ROM distribution and online databases, data should only be available to authorized users. To prevent unauthorized users from accessing data, the data supplier will encrypt data and provide only the authorized users with personal keys to decrypt it. However, some unauthorized users (*pirates*) may obtain some decryption keys from a group of one or more authorized users (*traitors*). Then the pirate users can decrypt data that they are not entitled to. To prevent this, Chor, Fiat and Naor [2] proposed k-resilient traceability schemes which reveal at least one traitor when a pirate decoder is confiscated if there are at most k traitors. Their schemes are, however, very inefficient and non constructive. In "open one level scheme", each user has to keep $O(k^2 \log n)$ personal decryption keys and the data supplier has to broadcast $O(k^4 \log n)$ ciphertexts, where n denotes the number of authorized users. Their other two schemes have very large keys and very long ciphertexts, similarly. Recently, Stinson and Wei showed some explicit constructions by using combinatorial designs [11]. Although their constructions may not be as good asymptotically as those in [2], they are often better for small values of k and n.

* A part of this work has been supported by JAIST.

** A part of this research has been supported by NSF Grant NCR-9508528.

On the other hand, this kind of traceability schemes are symmetric in the sense that the data supplier T is assumed to be honest. If T is dishonest, he can easily frame any authorized user as a traitor because T generates each user's key. Pfitzmann pointed out this problem and introduced asymmetric traceability schemes in which T cannot frame any user as a traitor [9]. She and Waidner [9, 10] showed an asymmetric scheme by combining the symmetric scheme of [2] with a two party protocol [1, 7]. This scheme is, however, not efficient because the symmetric scheme of [2], on which it is based, is very inefficient.

This paper first derives lower bounds on the sizes of keys and ciphertexts for symmetric traceability schemes. These bounds are all tight because an optimum one-time use scheme is also presented. Further, we show that optimum (k, n)-traceability schemes have a strong connection with orthogonal arrays. An upper bound on the number of authorized users is obtained as a corollary.

We then propose a multiple-use traceability scheme. It has much smaller keys and much shorter ciphertexts than the schemes by Chor, Fiat and Naor [2]. Our scheme requires one personal decryption key, $O(k)$ many encryption keys and $O(k)$ many ciphertexts. Further, our scheme is proven to be secure in the sense that (1) it satisfies secrecy requirement against outside enemies if and only if ElGamal cryptosystem is secure and (2) it can trace traitors if and only if the discrete log problem is hard. Further, the encryption key e_T of a data supplier can be made public. This means that everybody can work as a data supplier by using this public key e_T. The scheme of [2] does not have this property.

Our multiple-use scheme uses similar mathematics to threshold cryptosystems such as [4, 8]. A major difference is that a single user decrypts in our scheme while k users have to cooperate to decrypt in [4, 8]. These two concepts do not seem to be equivalent because while there exists an RSA type threshold cryptosystem [5, 3], we do not know how to make an RSA type traceability scheme with the public key property mentioned above. (We constructed an RSA type non-public key version, though. The details will be given in the final paper.)

Finally, we show two practical asymmetric traitor tracing schemes with agents or with an arbiter. The first is a multiple-use asymmetric scheme which contains c agents who only generate keys cooperatively. In this scheme, (1) no authorized user can be framed as a traitor even if T and $c-1$ agents collude and (2) T can detect a traitor and convince a judge without the help of agents. This scheme is computationally secure. The second is a one-time use asymmetric scheme with an arbiter which is unconditionally secure.

2 Preliminaries

2.1 Model and notation

In the model of symmetric traceability schemes, there are $n+2$ participants, a data supplier T, a set of n authorized users and a pirate user. T generates his encryption key e_T and a personal decryption key e_i for each authorized user i. To send actual plaintext data m only to authorized users, T first chooses a session key s. Then T broadcasts $(e_T(s), ENC_s(m))$, where $h \triangleq e_T(s)$ is called a *header* and ENC is a symmetric key encryption function.

Every authorized user i can recover s from $e_T(s)$ by using his personal key e_i and then decrypt $ENC_s(m)$ to obtain plaintext data m.

2.2 Previous traceability scheme

This subsection describes the basic k-resilient traceability schemes of Chor, Fiat and Naor [2]. All e_i are somehow constructed together with e_T as follows. Let $A_i \triangleq \{a_{i,1}, \ldots, a_{i,v}\}$ for $i = 1, 2, \ldots, l$, where $a_{i,j}$ is a random number. They are called *base keys*. Then T's encryption key is $e_T = A_1 \cup A_2 \cup \cdots \cup A_l$. The personal decryption key of authorized user i is $e_i = \{b_1, \ldots, b_l\}$, where b_j is randomly selected from A_j for $j = 1, 2, \ldots, l$. To encrypt a session key s, T first chooses random numbers $s_1, \ldots, s_l$ such that $s = s_1 + \cdots + s_l$. T then computes the header $h = e_T(s)$ as $h = B_1 \cup B_2 \cup \cdots \cup B_l$, where $B_i \triangleq \{K_{a_{i,1}}(s_i), \ldots, K_{a_{i,v}}(s_i)\}$ and K is a symmetric key encryption function. It can be seen that authorized users can decrypt each s_i and then obtain s. Now some malicious users (traitors) may conspire and give an unauthorized user a *pirate decoder* e_p. The pirate decoder e_P will consist of l base keys such that

$$e_p = \{\hat{b}_1, \ldots, \hat{b}_l\} \subseteq \bigcup_{i \in C} e_i, \qquad (1)$$

where C is the coalition of traitors such that $|C| \leq k$. The goal is to find at least one traitor of C. From eq.(1), we see that there exists at least one $u \in C$ such that $|e_p \cap e_u| \geq l/k$. Therefore, T finds an authorized user u such that $|e_p \cap e_u| \geq |e_p \cap e_i|$ for any $i \neq u$. This user u is defined as an exposed user.

As can be seen from the above, the sizes of keys and headers are very large. Their schemes are not constructive, either. More details will be surveyed in Sec.3.5.

3 Lower bounds

3.1 Definition

For a traceability scheme, let $\mathcal{E}_T$, $\mathcal{E}_i (i = 1, 2, \ldots, n)$, $\mathcal{S}$ and $\mathcal{HEAD}$ denote the random variables induced by e_T, e_i, s and h, respectively, where e_T is the data supplier's encryption key, e_i is the personal decryption key of user i, s is the session key, h is the header and n denotes the number of authorized users. For simplicity, we treat the key e_T as an encryption function as well, so $h = e_T(s)$. Similarly we view the personal keys e_i as decryption functions, so $s = e_i(h)$.

For a random variable $\mathcal{X}$, $H(\mathcal{X})$ denotes the entropy of $\mathcal{X}$. For $\mathcal{X}$, let $X \triangleq \{x \mid \Pr(\mathcal{X} = x) > 0\}$. $|X|$ denotes the cardinality of X. For simplicity, we assume that $|E_1| = |E_2| = \cdots = |E_n|$.

We assume that at most k authorized users are malicious.

Definition 1. We say that $(\mathcal{E}_T, \mathcal{E}_1, \ldots, \mathcal{E}_n, \mathcal{S}, \mathcal{HEAD})$ is a (k, n)-one-time traceability scheme if
(1) any outside enemy has no information on s from h. That is,

$$H(\mathcal{S} \mid \mathcal{HEAD}) = H(\mathcal{S}).$$

(2) Each authorized user i can uniquely decrypt s from h. That is,

$$H(\mathcal{S} \mid \mathcal{HEAD}, \mathcal{E}_i) = 0 \text{ for } 1 \leq \forall i \leq n.$$

(3) For every coalition of at most k authorized users (traitors), the following holds: Suppose that they (the traitors) use their personal decryption keys to construct a pirate decoder. If this decoder is capable of applying the decryption scheme, then one of the coalition members is identified with probability more than $1/|E_i|$.
(4) $\mathcal{S}$ and $(\mathcal{E}_1, \ldots, \mathcal{E}_n)$ are independent.

3.2 Lower bound on $|E_i|$

Theorem 2. *In a (k, n)-one-time traceability scheme,*

$$|E_i| \geq |S| \text{ for any } i. \tag{2}$$

Further, for any $h \in H$, define a mapping $v_h : E_i \to S \cup \{\perp\}$ as

$$v_h(e_i) \triangleq \begin{cases} s & \text{if } e_i(h) = s \\ \perp & \text{otherwise.} \end{cases} \tag{3}$$

Then v_h is a bijection from E_i to S if $|E_i| = |S|$.

Proof. Fix a header $h \in H$ arbitrarily. Let $v_h(E_i) \triangleq \{v_h(e_i) \mid e_i \in E_i\}$. Suppose that $s_0 \notin v_h(E_i)$ for some $s_0 \in S$. Then an outside enemy knows that $s_0 \notin v_h(E_i)$ from h. This is against Definition 1(1). Therefore, $v_h(E_i) \supseteq S$. This means that $|E_i| \geq |v_h(E_i)| \geq |S|$. It is clear that v_h is a bijection if $|E_i| = |S|$. □

The first part of Theorem 2 is originally due to Shannon. We presented a different proof in order to prove the second part which will be used in the proof of Theorem 5 and Theorem 11.

3.3 Lower bound on $|HEAD|$

Lemma 3. *In a (k, n)-one-time traceability scheme, no coalition C of at most k authorized users has any information on any other authorized user's key.*

Proof. Suppose that some coalition C of at most k users can guess the key e_u of some user $u \notin C$ with probability $p > 1/|E_u|$. Assume that C gives this e_u to a pirate as a pirate decoder. Then the data supplier will decide that user u is a traitor if he finds pirate decoder e_u. This happens with probability $p > 1/|E_u|$. This disagrees with Definition 1(3). □

Next, define $\hat{E}_{(i_1,\ldots,i_{k+1})} \subseteq E_{i_1} \times \cdots \times E_{i_{k+1}}$ as follows.

$$\hat{E}_{(i_1,\ldots,i_{k+1})} \triangleq \{(b_1,\ldots,b_{k+1}) \mid \Pr(\mathcal{E}_{i_1} = b_1,\ldots,\mathcal{E}_{i_{k+1}} = b_{k+1}) > 0\}.$$

Lemma 4. *If the equality of eq.(2) is satisfied for any i, then*

$$|\hat{E}_{(i_1,\ldots,i_{k+1})}| = |S|^{k+1} \quad \textit{for any } i_1,\ldots,i_{k+1}.$$

Proof. If $k = 0$, then this lemma is trivially true. For $k > 0$, choose $i_1,\ldots,i_{k+1}$ arbitrarily. From Lemma 3, user i_1 has no information on E_{i_2}. Therefore, for any $b_1 \in E_{i_1}$ and any $b_2 \in E_{i_2}$,

$$\Pr(\mathcal{E}_{i_2} = b_2 \mid \mathcal{E}_{i_1} = b_1) = \Pr(\mathcal{E}_{i_2} = b_2) > 0.$$

Hence, $\Pr(\mathcal{E}_{i_2} = b_2, \mathcal{E}_{i_1} = b_1) > 0$. Similarly, we have $\Pr(\mathcal{E}_{i_1} = b_1,\ldots,\mathcal{E}_{i_{k+1}} = b_{k+1}) > 0$ for any $(b_1,\ldots,b_{k+1}) \in E_{i_1} \times \cdots \times E_{i_{k+1}}$. Therefore,

$$|\hat{E}_{(i_1,\ldots,i_{k+1})}| = |E_{i_1}| \times \cdots \times |E_{i_{k+1}}| = |S|^{k+1}$$

since $\forall i : |E_i| = |S|$. □

Theorem 5. *In a (k,n)-one-time traceability scheme,*

$$\log|HEAD| \geq (k+1)\log|S| \tag{4}$$

if the equality of eq.(2) is satisfied for any i.

Proof. From Definition 1(4), any $(b_1,\ldots,b_{k+1}) \in \hat{E}_{(i_1,\ldots,i_{k+1})}$ can happen together with any $s \in S$, where $i_1,\ldots,i_{k+1}$ are arbitrary. This means that for any fixed s and for any $(b_1,\ldots,b_{k+1}) \in \hat{E}_{(i_1,\ldots,i_{k+1})}$, there exists some $h \in HEAD$ such that $b_1(h) = \cdots = b_{k+1}(h) = s$. Now fix $s_0 \in S$ arbitrarily. From the above observation, we can define a mapping $\gamma_{s_0} : \hat{E}_{(i_1,\ldots,i_{k+1})} \to HEAD$ as follows. For $(b_1,\ldots,b_{k+1}) \in \hat{E}_{(i_1,\ldots,i_{k+1})}$, choose $\hat{h} \in HEAD$ such that $b_1(\hat{h}) = \cdots = b_{k+1}(\hat{h}) = s_0$ arbitrarily. Then define

$$\gamma_{s_0}(b_1,\ldots,b_{k+1}) \triangleq \hat{h}. \tag{5}$$

Suppose that γ_{s_0} is not an injection. Then there exist some $(b_1,\ldots,b_{k+1})$ and $(b'_1,\ldots,b'_{k+1})$ such that

$$\gamma_{s_0}(b_1,\ldots,b_{k+1}) = \gamma_{s_0}(b'_1,\ldots,b'_{k+1}) = h_0$$

for some $h_0 \in HEAD$. Without loss of generality, suppose that $b_j \neq b'_j$. Since $|E_j| = |S|$, there exists the bijection $v_{h_0} : E_j \to S$ given by Theorem 2. Then we have $s_0 = b_j(h_0) \neq b'_j(h_0) = s_0$. This is a contradiction. Therefore, $\gamma_{s_0} : \hat{E}_{(i_1,\ldots,i_{k+1})} \to HEAD$ is an injection. Hence, $|HEAD| \geq |\hat{E}_{(i_1,\ldots,i_{k+1})}| = |S|^{k+1}$ from Lemma 4. □

3.4 Lower bound on $|E_T|$

Theorem 6. *Suppose that each $e_T \in E_T$ is a deterministic encryption function. If the equality of eq.(2) is satisfied for any i in a (k,n)-one-time traceability scheme, then*

$$\log|E_T| \geq (k+1)\log|S|. \tag{6}$$

Proof. For a fixed $s_0 \in S$, consider the mapping $\gamma_{s_0} : \hat{E}_{(i_1,\ldots,i_{k+1})} \to HEAD$ defined by eq.(5). Let $\gamma_{s_0}(\hat{E}_{(i_1,\ldots,i_{k+1})}) \triangleq \{h \mid h = \gamma_{s_0}(b_1,\ldots,b_{k+1}), (b_1,\ldots,b_{k+1}) \in \hat{E}_{(i_1,\ldots,i_{k+1})})\}$. Then $\gamma_{s_0}(\hat{E}_{(i_1,\ldots,i_{k+1})})$ is a subset of headers which can be generated by T for s_0. Therefore, we have

$$|E_T| \geq |\gamma_{s_0}(\hat{E}_{(i_1,\ldots,i_{k+1})})| = |\hat{E}_{(i_1,\ldots,i_{k+1})}| = |S|^{k+1}. \tag{7}$$

The first inequality holds since each e_T is deterministic. The first equality holds since γ_{s_0} is an injection as shown in the proof of Theorem 5. The last equality comes from Lemma 4. □

3.5 Comparison with CFN scheme

Chor, Fiat and Naor proposed three traceability schemes [2]. In these schemes,

$$\begin{aligned}
&\log|E_i| = 4k^2\log n\log|S|, && \log|HEAD| = 8k^4\log n\log|S|,\\
&\log|E_i| = 2k^2\log^2 k\log n\log|S|, && \log|HEAD| = 4k^3\log^4 k\log n\log|S|,\\
&\log|E_i| = 4k\log(n/p)/3\log|S|, && \log|HEAD| = 16k^2\log(n/p)/3\log|S|,
\end{aligned}$$

respectively, where p is the cheating probability of traitors. The above values are much larger than our bounds.

4 Optimum one-time use traceability scheme

4.1 Optimum scheme

In this subsection, we show an optimum (k,n)-one-time traceability scheme which meets all our bounds. This means that our bounds are all tight. (Also, the decryption needs polynomial time which Def.1 does not require.) Let $|S| = q$, where q is a prime and $q \geq n$.

Initialization: The data supplier T chooses a uniformly random polynomial $f(x) = a_0 + a_1x + \cdots + a_kx^k$ over $GF(q)$ as his encryption key e_T. Next, T gives $f(i)$ to authorized user i as a personal decryption key e_i for $0 \leq i \leq n-1$.

Distributing a session key: For a session key s, T broadcasts the header

$$h = (h_0, h_1, \ldots, h_k) = (s + a_0, a_1, \ldots, a_k)$$

which is computed over $GF(q)$. User i can compute s from h and e_i as follows.

$$(h_0 + h_1 \cdot i + \cdots + h_k \cdot i^k) - f(i) = s.$$

Detection of traitors: When a pirate decoder is confiscated, before an header is broadcast, the pirate key e_p is exposed. If e_p contains $(u, f(u))$ for some u, then T decides that user u is a traitor. (The scheme of [2] can detect traitors even if a pirate decoder is given as a blackbox. See [9, Sec.2.2].)

Definition 7. We say that a (k, n)-one-time traceability scheme is optimum if one has equalities in eq.(2), eq.(4) and eq.(6).

Theorem 8. *The above is an optimum (k, n)-one-time traceability scheme.*

Proof. It is clear that all the equalities of eq.(2), eq.(4) and eq.(6) are satisfied. We prove that Definition 1 is satisfied. Conditions (1), (2) and (4) are clear. Suppose that a coalition C of users $\{i_1, \ldots, i_k\}$ generates a pirate decoder e_p with probability more than $1/q$ such that e_p does not include $(i_1, f(i_1))$, or $(i_2, f(i_2))$, or ..., or $(i_k, f(i_k))$. Since e_p can decrypt s, e_p must contain at least $(x_0, f(x_0))$ for some $x_0 \notin \{i_1, \ldots, i_k\}$. However, this is impossible because $\deg f(x) = k$ and C knows only the k points of $f(x)$. □

4.2 Connection with orthogonal array

In this subsection, we show that optimum (k, n)-one-time traceability schemes have a strong connection with orthogonal arrays.

Definition 9. An *orthogonal array* $\mathrm{OA}(t, l, q)$ is a $q^t \times l$ array of q symbols such that in any t columns of the array, every possible t-tuple of symbols occurs in exactly one row. The parameter t is called the strength of the OA.

First, suppose that there exists an $\mathrm{OA}(k+1, k+n, q)$. Then we show that there exists a (k, n)-one-time traceability scheme such that $|S| = q$. Without loss of generality, assume that the set of symbols of the OA is $\{0, 1, \ldots, q-1\}$. Let $B = \{b_{i,j}\}$ denote the $q^{k+1} \times (k+n)$ matrix of the OA.

Initialization: The data supplier T chooses a row of B at random, say the $r(= e_T)$th row. T gives $b_{r,k+i}(= e_i)$ to user i as his personal key.

Distributing a session key: For a session key s, T broadcasts

$$h = (b_{r,1}, \ldots, b_{r,k}, b_{r,k+1} + s \bmod q)$$

as a header. User i decrypts s as follows. From $b_{r,1}, \ldots, b_{r,k}$ and $e_i = b_{r,k+i}$, he can determine r because the strength of the OA is $k+1$. Therefore, he can obtain $b_{r,k+1}$ and compute s.

Detection of traitors: When a pirate decoder is confiscated, the pirate key e_p is exposed. If e_p contains $(u, b_{r,k+u})$, then T decides that user u is a traitor.

Theorem 10. *If there exists an $OA(k+1, k+n, q)$, then there exists an optimum (k, n)-one-time traceability scheme such that $|S| = q$.*

A proof will be given in the final paper. Next, we show a weak converse.

Theorem 11. *If the equality of eq.(2) is satisfied for any i, then there exists an $OA(k+1, n, |S|)$.*

Proof. Let $ALL \triangleq \{(e_1, \ldots, e_n) \mid \Pr(\mathcal{E}_1 = e_1, \ldots, \mathcal{E}_n = e_n) > 0\}$. Consider a $|ALL| \times n$ matrix B which consists of all $(e_1, \ldots, e_n) \in ALL$. We show that this matrix B is an $\mathrm{OA}(k+1, n, |S|)$. Any $(e_1, \ldots, e_n) \in ALL$ can happen together with any $s \in S$ from Definition 1(4). Suppose that there exist $(e_1, \ldots, e_n) \in ALL$ and $(e_1, \ldots, e_{k+1}, e'_{k+2}, \ldots, e'_n) \in ALL$ such that $e_j \neq e'_j$ for some $k+2 \leq j \leq n$. For a fixed $s_0 \in S$, consider the mapping $\gamma_{s_0} : \hat{E}_{(1,\ldots,k+1)} \to HEAD$ defined by eq.(5). Let $\gamma_{s_0}(e_1, \ldots, e_{k+1}) = h_0$. Since $|E_j| = |S|$, there exists the bijection $v_{h_0} : E_j \to S$ given by Theorem 2. Then we have

$$s_0 = e_j(h_0) \neq e'_j(h_0) = s_0.$$

This is a contradiction. Therefore, $|ALL| = |\hat{E}_{(1,\ldots,k+1)}| = |S|^{k+1}$ from Lemma 4. The above discussion holds for $\forall(e_{i_1}, \cdots, e_{i_{k+1}})$. This means that B is an $\mathrm{OA}(k+1, n, |S|)$. □

Corollary 12. *If the equality of eq.(2) is satisfied for any i, then*

$$n \leq \begin{cases} |S| + k & \text{if } |S| \text{ is even and } k+1 \leq |S| \\ |S| + k - 1 & \text{if } |S| \text{ is odd and } 3 \leq k+1 \leq |S| \end{cases}$$

Proof. Apply Bush bound [12] to the OA of Theorem 11. We then obtain the desired bound. □

5 Multiple-use traceability scheme

In this section, we propose a multiple-use (k, n)-traceability scheme. It has much smaller keys and much shorter ciphertexts than those of CFN schemes [2]. Further, we prove that (1) our scheme satisfies computational secrecy if and only if ElGamal cryptosystem is secure and (2) that it satisfies computational traceability if and only if the discrete log problem is hard. Computational secrecy and computational traceability are defined as follows.

Definition 13. (1) Computational secrecy: No outside enemy can compute a session key s from a header h with nonnegligible probability, even after receiving many previous session keys.
(2) Computational traceability: No coalition C of at most k users can generate from their private keys and the public key, a pirate decoder e_p such that none of C is detected with nonnegligible probability.

In each case, an adversary is a probabilistic polynomial time Turing machine which can use the data supplier T as an oracle.

5.1 Computationally secure (k, n)-traceability scheme

We will combine our optimum (k, n)-one-time traceability scheme with ElGamal cryptosystem. Let p be a prime power. Let q be a prime such that $q \mid p-1$, $q \geq n+1$, and let g be a qth root of unity over $GF(p)$. All the participants agree on p, q and g. Let $S = \langle g \rangle = \{s \mid s = g^x \text{ for some } x\}$.

Initialization: T chooses a random polynomial $f(x) = a_0 + a_1 x + \cdots + a_k x^k$ over Z_q and computes

$$y_0 = g^{a_0}, y_1 = g^{a_1}, \ldots, y_k = g^{a_k}.$$

Let

$$e_T = (p, g, y_0, \ldots, y_k).$$

T gives $f(i)$ to authorized user i as e_i for $1 \leq i \leq n$.

Distributing a session key: For a session key s, T computes a header as

$$h(s, r) = (g^r, s y_0^r, y_1^r, \ldots, y_k^r),$$

where r is a random number. Then T broadcasts $h(s, r)$. Each user i computes s from $h(s, r)$ and e_i as follows.

$$\{s y_0^r \times (y_1^r)^i \times \cdots \times (y_k^r)^{i^k}\}/(g^r)^{f(i)} = s(g^r)^{f(i)}/(g^r)^{f(i)} = s.$$

Detection of traitors: When a pirate decoder is confiscated, the pirate key e_p is exposed. If e_p contains $(u, f(u))$ for some u, then T decides that user u is a traitor.

In our scheme, e_T can be made public. This means that everybody can work as a data supplier by using this public key e_T. We will prove, under reasonable assumptions, that our scheme is computationally secure even if e_T is a public key.

5.2 Computational secrecy

Theorem 14. *The computational complexity for an eavesdropper to cryptanalyze a new session key in our scheme, after having received previous session keys, the public key, the old headers and the new header is as hard (reducible in expected polynomial time) to cryptanalyze a plaintext in the ElGamal scheme when the order of g is a prime.*

Proof. Let $\mathcal{M}_1$ correspond with the problem of breaking our scheme and $\mathcal{M}_2$ with breaking the standard ElGamal encryption scheme when the order of g is a prime. First, it is clear that the existence of a polynomial time $\mathcal{M}_2$ implies the existence of a polynomial time $\mathcal{M}_1$. Secondly, suppose that there exists a polynomial time $\mathcal{M}_1$. We will show $\mathcal{M}_2$ by using $\mathcal{M}_1$ as a subroutine. Let the input to $\mathcal{M}_2$ be (p, g, y), (α_0, β_0) $(= (g^{r_0}, s_0 y^{r_0}))$, $\ldots$, (α_l, β_l) $(= (g^{r_l}, s_l y^{r_l}))$.

$\mathcal{M}_2$ computes e_T and $h(s_j, r_j)$ which will be used as an input to $\mathcal{M}_1$ as follows. $\mathcal{M}_2$ first chooses $a_1, \ldots, a_k \in Z_q$ at random and computes

$$y_1 = g^{a_1}, \ldots, y_k = g^{a_k}.$$

Then $\mathcal{M}_2$ has obtained a public key $e_T = (p, g, y, y_1, \ldots, y_k)$. Next, $\mathcal{M}_2$ computes $y_1^{r_j}, \ldots, y_k^{r_j}$ as follows.

$$\alpha_j^{a_i} = g^{r_j a_i} = y_i^{r_j} \text{ for } i = 1, 2, \ldots, k,\ j = 0, 1, \ldots, l.$$

Thus, $\mathcal{M}_2$ has obtained $h(s_j, r_j)$ such that

$$h(s_j, r_j) = (\alpha, \beta, \alpha^{a_1}, \ldots, \alpha^{a_k}) = (g^{r_j}, s_j y^{r_j}, y_1^{r_j}, \ldots, y_k^{r_j}).$$

Now, $\mathcal{M}_2$ feeds the above e_T and $h(s_j, r_j)$ $(j = 0, 1, \ldots, l)$ to $\mathcal{M}_1$. Next, for any $s_i \in S$ queried by $\mathcal{M}_1$, $\mathcal{M}_2$ computes $h(s_i, r_i)$ by using e_T and sends $h(s_i, r_i)$ to $\mathcal{M}_1$. Note that it is easy to compute $h(s_i, r_i)$ by using e_T for any s_i.

Finally, $\mathcal{M}_2$ outputs the output of $\mathcal{M}_1$, which is an s_j with nonnegligible probability. Thus, the existence of $\mathcal{M}_1$ implies the existence of $\mathcal{M}_2$. □

5.3 Computational traceability

Theorem 15. *The computational complexity for k traitors of finding a pirate decoder $(u, f(u))$, where $u \notin \{i_1, \ldots, i_k\}$, when given the public key and their personal keys $(f(i_1)), \ldots, f(i_k))$ is as hard as (reducible in expected polynomial time to) the discrete logarithm problem when the order of g is a prime.*

Proof. We call $\mathcal{M}_1$ the algorithm that k traitors would use to find a pirates decoder and $\mathcal{M}_2$ an algorithm to solve the discrete logarithm problem when the order of g is a prime. First, it is clear that the existence of a polynomial time $\mathcal{M}_2$ implies the existence of a polynomial time $\mathcal{M}_1$. Secondly, suppose that there exists a polynomial time $\mathcal{M}_1$. We show a polynomial time $\mathcal{M}_2$ by using $\mathcal{M}_1$ as a subroutine. Let the input to $\mathcal{M}_2$ be (p, g, y). $\mathcal{M}_2$ first chooses $d_1, \ldots, d_k$ at random. Then there exists a unique polynomial $f(x) = a_0 + a_1 x + \cdots + a_k x^k$ such that

$$y = g^{a_0} \text{ and } f(i_j) = d_j \text{ for } 1 \le j \le k.$$

Now $\mathcal{M}_2$ computes $y_j = g^{a_j}$ $(j = 1, 2, \cdots, k)$ as follows. It holds that

$$(d_1, \cdots, d_k)^T = (f(i_1), \cdots, f(i_k))^T = (a_0, \cdots, a_0)^T + B \times (a_1, \cdots, a_k)^T,$$

where

$$B \triangleq \begin{pmatrix} i_1, & i_1^2, & \cdots, & i_1^k \\ \vdots & \vdots & \vdots & \vdots \\ i_k, & i_k^2, & \cdots, & i_k^k \end{pmatrix}.$$

B is nonsingular because it is a Vandermonde matrix. Therefore, we have

$$(a_1, \cdots, a_k)^T = B^{-1}(d_1 - a_0, \cdots, d_k - a_0)^T.$$

Let $(b_{j1}, \cdots, b_{jk})$ be the jth row of B^{-1}. Then

$$\begin{aligned} a_j &= b_{j1}(d_1 - a_0) + \cdots + b_{jk}(d_k - a_0), \\ &= b_{j1} d_1 + \cdots + b_{jk} d_k - (b_{j1} + \cdots + b_{jk}) a_0. \end{aligned}$$

Hence,

$$g^{a_j} = g^{b_{j1}d_1+\cdots+b_{jk}d_k}/y^{b_{j1}+\cdots+b_{jk}}.$$

for $j = 1, 2, \cdots, k$. Now, $\mathcal{M}_2$ has obtained a public key $e_T = (p, g, y, g^{a_1}, \ldots, g^{a_k})$ and personal keys of traitors $d_1 = f(i_1), \ldots, d_k = f(i_k)$.

$\mathcal{M}_2$ feeds e_T and $(i_1, d_1), \ldots, (i_k, d_k)$ to $\mathcal{M}_1$. Finally, if $\mathcal{M}_1$ outputs $(u, f(u))$ such that $u \notin \{i_1, \ldots, i_k\}$, then $\mathcal{M}_2$ can compute $f(x)$ from $(i_1, d_1), \ldots, (i_k, d_k)$ and $(u, f(u))$. In this case, $\mathcal{M}_2$ outputs $a_0 = f(0)$ which is the discrete log of y. This happens with nonnegligible probability. Otherwise, $\mathcal{M}_2$ outputs $\perp$. Thus, the existence of $\mathcal{M}_1$ implies the existence of $\mathcal{M}_2$. □

6 Asymmetric schemes with agents/arbiter

Up to now, we have considered symmetric schemes in which the data supplier T is honest. If T is dishonest, he can easily frame any authorized user as a traitor because T knows all the personal keys. To solve this problem, Pfitzmann and Waidner [9, 10] showed an asymmetric scheme by combining the symmetric scheme of [2] with a two party protocol [1, 7]. This scheme is, however, not efficient because at least the symmetric scheme of [2] which it is based on is very inefficient.

In this section, we show two practical asymmetric schemes with agents/arbiter.

6.1 Computationally secure asymmetric scheme with agents

This is a multiple-use asymmetric scheme in which there are c agents $\mathcal{A}_1, \ldots, \mathcal{A}_c$ who only generate keys cooperatively. In this scheme,
(1) no user is framed as a traitor even if T and $c-1$ agents collude.
(2) When a pirate decoder is confiscated, T can detect a traitor without any help from agents.
(3) At a trial, T can convince a judge without the help of agents.

This scheme is obtained by modifying the scheme of Sec.5.1 using the same mathematics as in [8]. All the participants agree on p and g.

Initialization: Each agent $\mathcal{A}_i$ chooses a random polynomial $f_i(x) = a_{i,0} + a_{i,1}x + \cdots + a_{i,k}x^k$ over Z_q and publicizes

$$y_{i,0} = g^{a_{i,0}}, \quad \ldots, \quad y_{i,k} = g^{a_{i,k}}.$$

Each $\mathcal{A}_i$ secretly gives $f_i(j)$ to user j. Let

$$f(x) \triangleq \sum_{i=1}^{c} f_i(x) = a_0 + a_1 x + \cdots a_k x^k, \; y_j \triangleq \prod_{i=1}^{c} y_{i,j} \text{ for } 0 \leq j \leq k.$$

Then $y_j = g^{a_j}$ for $1 \leq j \leq k$. T uses $(p, g, y_0, \ldots, y_k)$ as a public key e_T. Authorized user j computes $f(j) = \sum_{i=1}^{c} f_i(j)$ which he uses as his personal key e_j, where $1 \leq j \leq n$.

The phase for *Distributing a session key* is the same as that in Sec.5.1.

Detection of traitors: When a pirate decoder is confiscated, the pirate key e_p is exposed. Suppose that e_p contains $(u, f(u))$. T decides that user u is a traitor if

$$g^{f(u)} = \prod_{i=0}^{k} y_i^{u^i}. \tag{8}$$

Trial: If T decides that user u is a traitor, then T gives the judge $(u, f(u))$ as evidence. The judge is convinced if eq.(8) holds.

Theorem 16. *T and $c-1$ agents cannot frame any authorized user as a traitor with nonnegligible probability if and only if the discrete log problem is hard.*

A proof and a straightforward use of [8] to achieve verifiability of the correctness of $f(i)$, will be given in the final paper.

6.2 Unconditionally secure asymmetric scheme with arbiter

This is a one-time use asymmetric scheme in which there is an arbiter $\mathcal{A}$ who generates e_T and $\{e_i\}$. When a pirate decoder is confiscated, $\mathcal{A}$ detects a traitor. This scheme is obtained by modifying the scheme in Sec.4.1 as follows. Let $|S| = q$, where q is a prime and $q \geq n$.

Initialization: An arbiter $\mathcal{A}$ gives the data supplier T two random polynomials

$$f_1(x) = a_{1,0} + a_{1,1}x + \cdots + a_{1,k}x^k, \ f_2(x) = a_{2,0} + a_{2,1}x + \cdots + a_{2,k}x^k.$$

over GF(q) as e_T. Next, $\mathcal{A}$ gives $(x_{i,1}, x_{i,2}, f_1(x_{i,1}), f_2(x_{i,2}))$ to authorized user i as e_i, where each $x_{i,j}$ is independently and randomly chosen from $GF(q)$.

Distributing a session key: For a session key s, T chooses two random elements s_1 and s_2 such that $s = s_1 + s_2$. Then T broadcasts (h_1, h_2) such that

$$h_i = (s_i + a_{i,0}, a_{i,1}, \ldots, a_{i,k}), \quad \text{where } i = 1, 2$$

as a header. User i computes s_1 and s_2 as in Sec.4.1 and then obtains s.

Detection of traitors: When a pirate decoder is confiscated, the pirate key e_p is exposed. Suppose that e_p contains $(x_{u,1}, x_{u,2}, f_1(x_{u,1}), f_2(x_{u,2}))$. Then the arbiter decides that user u is a traitor.

Theorem 17. *T cannot frame any user as a traitor with probability more than $1/q$.*

Proof. Suppose $\mathcal{A}$ gave e_T to T and e_i to authorized user i for $1 \leq i \leq n$. Then it is clear that $|\{e_i\}| = n$. On the other hand, T has no information on $x_{i,j}$. Therefore, Pr(T can frame some *user*) $= n/q^2 \leq 1/q$ since $q \geq n$. □

7 Open problem

The bounds given in this paper only apply to the unconditionally secure case. Can these be adapted to a public key scenario?

References

1. D. Chaum, I.B. Damgard, and J. Graaf. "Multiparty computations ensuring privacy of each party's input and correctness of the result". In *Proc. of Crypto'87, Lecture Notes in Computer Science, LNCS 293, Springer Verlag*, pages 87–119, 1988.
2. B. Chor, A. Fiat, and M. Naor. "Tracing traitors". In *Proc. of Crypto'94, Lecture Notes in Computer Science, LNCS 839, Springer Verlag*, pages 257–270, 1994.
3. A. De Santis, Y. Desmedt, Y. Frankel, and M. Yung. "How to share a funciton securely ". In *Proc. of STOC'94*, pages 522–533, 1994.
4. Y. Desmedt and Y. Frankel. "Threshold cryptosystems". In *Proc. of Crypto'89, Lecture Notes in Computer Science, LNCS 435, Springer Verlag*, pages 307-315, 1990.
5. Y. Desmedt and Y. Frankel. "Homomorphi zero-knowledge threshold schemes over any finite Abelian group ". In *SIAM J. on Discrete Math., vol.7, no.4*, pages 667–679, 1994.
6. Y. Desmedt, Y. Frankel, and M. Yung. Multi-receiver / multi-sender network security: efficient authenticated multicast/ feedback. In *IEEE INFOCOM '92*, pp. 2045–2054, 1992.
7. O. Goldreich, S. Micali, and A. Wigderson. "How to Play Any Mental Game". In *Proc. of the 19th ACM Symposium on Theory of Computing*, pages 218–229, 1987.
8. T. Pedersen. "A threshold cryptosystem without a trusted party". In *Proc. of Eurocrypt'91, Lecture Notes in Computer Science, LNCS 547, Springer Verlag*, pages 522-526, 1991.
9. B. Pfitzmann. "Trials of traced traitors". In *Proc. of Information Hiding, Lecture Notes in Computer Science, LNCS 1174, Springer Verlag*, pages 49–64, 1996.
10. B. Pfitzmann and M. Waidner. "Asymmetric fingerprinting for larger collusions". In *4th ACM Conference on Computer and Communication Security*, 1997.
11. D. Stinson and R. Wei. "Combinatorial properties and constructions of traceability schemes and frameproof codes". In *SIAM J. on Discrete Math., vol.11, no.1*, pages 41–53, 1998.
12. "The CRC handbook of combinatorial designs". In *CRC press inc.,* 1996.