A Flexible-Revocation Scheme for Efficient Public-Key Black-Box Traitor Tracing

Tatsuyuki MATSUSHITA*,†, Member and Hideki IMAI††, Fellow

SUMMARY We propose a new type of revocation scheme for efficient public-key black-box traitor tracing. Our revocation scheme is flexible and efficient in the sense that (i) any number of subscribers can be revoked in each distribution under an assumption that the number of revoked subscribers who collude in one coalition is limited to a threshold and (ii) both each subscriber’s storage and the transmission overhead are independent of \( n \), while (i) the maximum number of revoked ones cannot be changed or (ii) they depend on \( n \) in previous schemes, where \( n \) is the total number of subscribers. The flexibility in revocation is significant since flexible revocation can be integrated with efficient black-box tracing and this integration can be achieved without a substantial increase in the transmission overhead over the previous schemes. In this paper, we present a concrete construction of an efficient public-key black-box traceable and revocable scheme by combining flexible revocation with a known black-box tracing algorithm which works under the same attack model as assumed in the previous schemes. Our scheme achieves that (i) the transmission overhead remains efficient, especially linear only in \( k \) in case of bulk revocation and (ii) the tracing algorithm runs in \( O(\log n) \) time, while the previous ones cannot satisfy both of these properties, where \( k \) is the maximum number of traitors in a coalition.

key words: public-key traitor tracing, black-box tracing, flexible revocation

1. Introduction

Consider the following content-distribution system: A data supplier distributes digital contents (e.g., a movie) to subscribers over a broadcast channel. Since the contents should be available only to subscribers, the data supplier broadcasts the encrypted contents. Only subscribers can decrypt them with their corresponding decryption keys given in advance. This system can be applied to pay-TV, DVD-ROM distribution, online database, etc. In this content-distribution system, malicious subscribers may redistribute their decryption keys to non-subscribers. This piracy should be prevented since it allows the non-subscribers to have illegal access to the contents.

As a deterrent to the piracy, traitor tracing has been the subject of extensive studies. The concept of traitor tracing was introduced by Chor-Fiat-Naor [2]. In traitor tracing, each subscriber is given a distinct decryption key (personal key) which is contained in a decryption device (decoder), and the data supplier broadcasts both the contents encrypted with a session key and the encrypted session key (header). Typically, a symmetric key is used as the session key. The subscribers can obtain the session key (and consequently the contents) by inputting the received header to their decoders. In this scenario, malicious subscribers (traitors) may give away their personal keys to a pirated version of a decoder (pirate decoder). Once the pirate decoder is found, at least one of the traitors who join the piracy can be identified from it. A traitor-tracing scheme discourages traitors from committing the piracy since the confiscated pirate decoder can be traced back to its producers.

Traitor-tracing schemes can be classified into three types according to their constructions: combinatorial, tree-based, and algebraic ones. The first type of scheme [2]–[4] is inefficient in the following criteria: each subscriber’s storage and the transmission overhead. This is because it has to greatly degrade the efficiency in order to eliminate the probability that an honest subscriber is falsely detected as a traitor. In the other types of scheme, the above dilemma is removed. The second type of scheme [5]–[7] requires each subscriber to store a unique and shared personal keys the number of which is logarithmic in the total number of subscribers. It has a desirable property that traitors cannot compute (with non-negligible probability) all of the personal keys assigned to a subscriber even if the other ones collude. However, if a traitor builds the pirate decoder in which only the shared personal keys are embedded, the pirate decoder cannot be traced back to the traitor with overwhelming probability, although all of the embedded personal keys can be revoked. The third type of scheme [8], [9] is efficient in the sense that each subscriber’s storage is constant (e.g., 160 bits) and the transmission overhead is linear only in the maximum number of traitors in a coalition. The scheme of [8] is constructed by modifying ElGamal cryptosystem so that there is one public key for all of the subscribers and the subscribers can decrypt the header encrypted with the public key by using their personal keys. The schemes of [8] and [9] are closely related. In fact, it is pointed out in [10] that the schemes of [9] is equivalent to a modified version of [8]. In this type of scheme, the above traceability problem can be avoided, though there is a threshold of the maximum coalition size. Our interest is in this type of scheme.

Among efficient public-key traitor-tracing schemes, the schemes of [11]–[13] achieve revocation of subscribers.
These schemes make use of the key-distribution method proposed in [14], which can viewed as a one-move Diffie-Hellman key exchange with revocation. In [14], revocation is achieved by performing Shamir’s secret sharing in the exponents. In the schemes of [11]–[13], the data supplier can revoke a certain number of subscribers in each distribution, i.e., the data supplier can make their decoders useless without confiscating them. The schemes are suitable for the content-distribution system where the subscribers frequently cancel/renew their subscriptions to the contents. Moreover, the schemes have another desirable property: the revocation mechanism can also be used to support black-box tracing, which provides the assurance that traitors can be identified no matter how the pirate decoder is implemented. In black-box tracing, a tracer does not break open the pirate decoder but uses it as a black box. Briefly, the tracer chooses suspects and tests whether traitors are among them only by observing the behavior of the pirate decoder on chosen inputs. If revocation is supported, the tracer can perform the above test by revoking the suspects in the input. Therefore, the revocation mechanism can be integrated seamlessly with black-box tracing.

Unfortunately, in the previous schemes [11]–[13], the revocation mechanism cannot be integrated with efficient black-box tracing, i.e., the black-box tracing algorithm of the previous schemes has to be exponential, hence impractical, in order to keep the transmission overhead efficient. This trade-off greatly spoils the scalability of the schemes. The scheme of [7], which is a public-key extension of [5], supports efficient black-box tracing but both each subscriber’s storage and the transmission overhead depend on the total number of subscribers.

In this paper, we solve the above efficiency problem by getting rid of a trade-off between the transmission overhead and the running time of the tracing algorithm. Our scheme satisfies the following properties:

**Flexible revocation** While in the previous schemes (i) the number of revoked subscribers is limited to a certain threshold which cannot be changed unless the system is initialized again or (ii) each subscriber’s storage and the transmission overhead depend on the total number of subscribers, in our scheme, (i) the upper bound on this parameter can be set arbitrarily in each distribution without any reinitialization and (ii) they do not. Note that even though there is a threshold of the coalition size of traitors, revocation of any number of subscribers can be achieved in the following case we consider in the paper: A set of revoked subscribers includes one or more disjoint subsets, $X_1, \ldots, X_t$. Each $X_j$ is a distinct coalition in which revoked ones collude and $|X_j|$ is limited to the threshold. Our main contribution is providing a revocation mechanism which can drastically reduce the running time of the tracing algorithm without a substantial increase in the transmission overhead over the previous schemes. Thanks to flexible revocation, our scheme can achieve the efficient transmission overhead and efficient running time of the tracing algorithm, while the previous ones cannot accomplish both.

**Efficient black-box tracing** For completeness of our scheme, we give an explicit description of the black-box tracing algorithm mentioned in [12]. Note that the improvement we achieve is not in the tracing algorithm itself but in the efficient integration between tracing and revocation. The tracing algorithm can detect at least one of the traitors in a coalition with running time $O(\log n)$, while the running time of the previous one is $O\left(\binom{n}{t}\right)$, where $n$ is the total number of subscribers and $k$ is the maximum coalition size. The tracing algorithms can work under the same threat model as assumed in the previous schemes.

**Public-key setting** Similar to the previous schemes, anyone can work as a data supplier and/or a tracer in our scheme since no secret information is needed to build the header and to execute the tracing algorithm. This property is desirable because of the following two reasons: (i) it enhances the sender-scalability in the sense that plural data suppliers can use the same system and (ii) it provides public scalability of the tracing result, which is a stronger deterrent to the piracy.

Although our scheme achieves flexible revocation and efficient public-key black-box tracing, the computational cost for decryption remains to be improved. In our scheme, the number of exponentiations required for decryption increases as the maximum coalition size is set to be larger.

In Sect. 2, the assumption on the pirate decoder is described. We propose an efficient public-key black-box traceable and revocable scheme in Sect. 3. The proposed scheme is analyzed in terms of security and efficiency in Sect. 4 and Sect. 5, respectively. We present our conclusions in Sect. 6.

2. Model of Pirate Decoders

We adopt the same assumption on the pirate decoder as in the previous schemes.

**Assumption 1** The pirate decoder always outputs the correct plaintext if it gets the header of a regular form, i.e., it does not take measures that might fool the tracer.

We mean by the header of a regular form that the header is indistinguishable from the one for the normal broadcast. It is possible for traitors to construct a pirate decoder that can take measures that fool the tracer whatever it receives as the header. However, such a decoder could hardly be an item for sale. For example, consider that the digital data of a movie is distributed. The data is divided into 3-second lengths, and each part of the data is encrypted with a different session key. Traitors can build the pirate decoder which intentionally fails to calculate the session key every few seconds to escape from the analysis, but no one want to buy such a decoder because only fragments of the movie can be watched by using it. Therefore, Assumption 1 is not strong.

3. Proposed Scheme

To achieve both efficient revocation and efficient black-box
tracing, we combine the scheme of [14] and that of [8]. The main differences in the constructions between these schemes and ours are as follows: (i) In our scheme, a set of subscribers is split into disjoint subsets and a different key-generation polynomial is assigned to each subset while there is a single key-generation polynomial for all of the subscribers in [14] and [8], and (ii) our scheme can support revocation of any number of subscribers (though there is a threshold of the maximum coalition size) while the number of revoked subscribers in [14] is limited to a pre-determined threshold. Due to these differences, our scheme can achieve flexible revocation and therefore efficient black-box tracing without a substantial increase in the transmission overhead.

First, we describe an outline of the proposed scheme. Secondly, the explicit construction of our scheme is shown.

3.1 Outline

Our scheme consists of the four phases.

**Key generation:** A trusted party generates and secretly gives every subscriber a distinct personal key. The personal key is stored in the decoder.

**Encryption:** The data supplier encrypts (i) the digital contents with the session key and (ii) the session key itself as the header. In the header, the selected subscribers are revoked. Then, the data supplier broadcasts the encrypted digital contents and the header. To avoid complication, we assume that (i) the symmetric encryption algorithm, which is used for encryption of the contents, is secure and publicly known and (ii) a broadcast channel is reliable in the sense that the received information is not altered.

**Decryption:** When receiving the header, subscribers compute the session key by inputting it to their decoders. Since only non-revoked subscribers can compute the session key, the digital contents are available only to them.

**Tracing:** Suppose that the pirate decoder is confiscated. The tracer chooses a set of suspects and builds the header in which the selected suspects are revoked. The tracer inputs the chosen headers to the pirate decoder and observes whether it decrypts the headers correctly or not. If its output is (i) correct on the input where a set of revoked suspects is \( T \) and (ii) incorrect on the input where a set of revoked suspects is \( T \cup \{u\} \), then the tracer decides the subscriber, \( u \), is a traitor.

3.2 Protocol

Let \( n \) be the total number of subscribers and \( k \) be the maximum number of traitors in a coalition. Let \( p, q \) be primes s.t. \( q | p - 1 \) and \( q \geq n + k + 1 \). Let \( g \) be a \( q \)-th root of unity over \( \mathbb{Z}_p^* \) and \( G_q \) be a subgroup of \( \mathbb{Z}_p^* \) of order \( q \). Let \( \mathcal{U} \) be a set of subscribers \( \mathcal{U} \subseteq \mathbb{Z}_p \) and \( \mathcal{X} \) be a set of revoked subscribers. Let \( z \in \mathbb{R} \) denote a random selection of an element of the set on its right side. All the participants agree on \( p, q \), and \( g \). The calculations are done over \( \mathbb{Z}_p \) unless otherwise specified.

**Key generation:** Choose \( a_0, \ldots, a_k, b_1, \ldots, b_k \in \mathbb{Z}_q \). Then, compute the public key \( e \) as follows:

\[
e = (g, y, y_0, \ldots, y_{0,2}, y_{1,1}, \ldots, y_{1,k}) = (g, g^{a_0}, \ldots, g^{a_k}, g^{b_1}, \ldots, g^{b_k}).
\]

Split \( \mathcal{U} \) into \( k \) disjoint subsets \( \mathcal{U}_1, \ldots, \mathcal{U}_k \). These subsets are publicly known. Suppose that \( u \in \mathcal{U}_i \). The subscriber \( u \)'s personal key is \( (u, i, f(u)) \) where

\[
f_i(x) = \sum_{j=0}^{k} a_{i,j} x^j \mod q,
\]

\[
a_{i,j} = \begin{cases} a_j & (i \neq j), \\ b_j & (i = j). \end{cases}
\]

**Encryption:** Check whether \( \mathcal{Y} = \mathcal{X} \cup \bigcup_{j \in [n]} (\mathcal{U}_j \subseteq \mathcal{X}, z_j \neq 0) \) is a non-empty set or not. If \( \mathcal{Y} = \emptyset \) or \( \mathcal{X} = \emptyset \), set \( m \leftarrow (k + 1) + k \). Otherwise, if \( \mathcal{Y} \neq \emptyset \), set \( m \leftarrow k \) and \( w \leftarrow 0 \). Select \( c_0, \ldots, c_m \in \mathbb{R} \), and \( x_0, \ldots, x_m \in \mathbb{R} \) if \( w < m \). Then, build the header \( h(r, X) \) as follows:

\[
h(r, X) = (h, h_0, \ldots, h_{0,m}, h_{1,1}, \ldots, h_{1,m}, H_1, \ldots, H_m),
\]

where

\[
h = g^r,
\]

\[
h_{0,j} = y_{0,j}^r g^j,
\]

\[
X_j = j \mod (k + 1),
\]

\[
X_{i,j} = \begin{cases} g^{j} & (X_j \subseteq \mathcal{X}, z_j \neq 0), \\ y_{0,j}^r g^j & (X_j \subseteq \mathcal{X}, z_j \neq 0), \end{cases}
\]

\[
H_j = (x_j, g^{F(x_j)})
\]

\[
F(x) = \sum_{j=0}^{m} c_j x^j \mod q,
\]

and \( r, r_j \) for each \( j \in [n] \) \( 1 \leq z \leq m \), \( \mathcal{U}_j \mod (k + 1) \subseteq \mathcal{X} \) are random numbers generated by the data supplier. If \( z_j = 0 \), then \( H_1 \) is not included in the header. Each 2-tuple \( H_j \) is a distinct share of \( g^{F(x)} = g^a \), which is the session key.

**Decryption:** Suppose that \( x_0 \in \mathcal{U}_i \). If \( x_0 \notin \mathcal{X} \), the subscriber, \( x_0 \), can correctly compute one share of \( g^{F(x)} \), i.e., \( (x_0, g^{F(x)}) \) as follows:

\[
g^{F(x_0)} = D(x_0) \sum_{j=0}^{m} c_j g^{F(x_j)},
\]

where \( d = (m-k)/(k+1) \) and

\[
D(x_0) = \prod_{j=0}^{m} B_{i,j},
\]

\[
B_{i,j} = \begin{cases} h_{0,j} & (i \neq j \mod (k + 1)), \\ h_{1,j} & (i = j \mod (k + 1)). \end{cases}
\]

\[
D_i(x_0) \text{ can be calculated as follows:}
\]

\[
D_i(x_0) = \prod_{j=0}^{d} (h_{0,k+1} \times h_{0,k+1}^r \times \cdots \times
\]
Step 1. Label all of the elements in $\{x_0, g^{F(n)}\}$. Then, compute the session key, $g^{F(0)}$, by performing the Lagrange interpolation in the exponents:

$$
g^{F(0)} = \prod_{j=0}^{m} \left( g^{F(x_j)} \right)^{L_j}
= g^{x_0 L_1} \times g^{x_d U_0},
$$

where

$$L_j = \prod_{0 \leq i \neq j < m} \frac{x_i}{x_i - x_j} \mod q.$$

**Tracing:** We describe how a known efficient black-box tracing algorithm is applied to our scheme. It runs in $O(\log n)$ time and can detect at least one of the traitors who are responsible for the confiscated pirate decoder by using binary search.

**Algorithm 1 (Binary-search black-box tracing)**

Input: $U_1, \ldots, U_k, \text{ and the pirate decoder, } D$.

Output: a traitor’s ID.

Step 1. Label all of the elements in $U_1, \ldots, U_k$ as follows:

$$U_1 = \{u_1, \ldots, u_d\},$$
$$U_2 = \{u_{d+1}, \ldots, u_{d+d_2}\},$$
$$\vdots$$
$$U_k = \{u_{d_{k-1}+1}, \ldots, u_{d_{k-1}+d_k}\}.$$

Step 2. Set $L_0 \gets 0, H_i \gets n, \ell \gets 1$. For $1 \leq \ell \leq \lceil \log_2 n \rceil$, repeat the following procedures:

1. Set $M_i \gets \lceil (L_i + H_i)/2 \rceil$ and $T_i \gets \{u_1, \ldots, u_{M_i}\}$. Then, execute the algorithm $A_{\text{test}}$ (described in Algorithm 2) with input $U_1, \ldots, U_k, T_i, D$.
   - If $A_{\text{test}}(U_1, \ldots, U_k, T_i, D) = \text{“correct”}$, then set $L_i \gets M_i.$
   - Otherwise (the output is “incorrect”), set $H_i \gets M_i.$

2. Increment $\ell$ by one and go to (2-1).

Step 3. After the $\lceil \log_2 n \rceil$ tests, find $u \in U$ which satisfies the following two conditions for some $\ell \in \{1, \ldots, \lceil \log_2 n \rceil\}$:

$$A_{\text{test}}(U_1, \ldots, U_k, T_\ell, D) = \text{“correct”},$$

Now, the subscriber, $x_0$, obtains the $m+1$ shares $H_1, \ldots, H_m$, and $(x_0, g^{F(n)})$. Then, performing the Lagrange interpolation in the exponents:

$$g^{F(0)} = \prod_{j=0}^{m} \left( g^{F(x_j)} \right)^{L_j}
= g^{x_0 L_1} \times g^{x_d U_0},$$

Step 2. Input $h(r, T_\ell)$ to $D$ and observe its output.

- If $D$ outputs the correct session key, $g^{F(0)}$, on the input, then output “correct.”
- Otherwise (the target is not even there), output “incorrect.”

We note that it is possible for the tracer to identify more than one of the traitors by repeating Algorithm 1 with differently permuted $U_1, \ldots, U_k$. For example, the tracer can relabel $U_1, \ldots, U_k$ as $V_1, \ldots, V_k$ in random order and label all of the elements of each $V_i$ in random order.

**4. Security**

The security of our scheme is based on the difficulty of the Decision Diffie-Hellman problem (DDH) [15]. Informally, the assumption that DDH in $G_q$ is intractable means that no polynomial-time algorithm can distinguish with non-negligible advantage between the two distributions $\langle g_1, g_2, g_1^b, g_2^b \rangle$ and $\langle g_1, g_2, g_1^b, g_2^b \rangle$ where $g_1, g_2 \in R G_q$, and $a, b \in R Z_q$. We call a 4-tuple coming from the former distribution as a Diffie-Hellman tuple.

4.1 Secrecy

Let $C$ be a set of revoked subscribers in a coalition and recall that $X$ is a set of revoked subscribers.

**Lemma 1** Suppose that the subscribers in $X$ are revoked after obtaining a certain number (bounded by a polynomial) of previous session keys and headers, the new header, the public key, and their personal keys. For any $X$, the computational complexity for any coalition of $k$ revoked subscribers to distinguish the session key corresponding to the new header from a random element in $G_q$ is as difficult as DDH in $G_q$. 

Proof Let $M_{c,x}^{\text{dist}}$ be a polynomial-time algorithm the coalition $C$ uses to distinguish the session key corresponding to the new header from a random element in $G_q$ when a set of revoked subscribers is $X$. Let $M_{c,x}^{\text{DDH}}$ be a polynomial-time algorithm which solves DDH in $G_q$. For two polynomial-time algorithms $M_0, M_1$, we mean by $M_0 \Rightarrow M_1$ that the existence of $M_0$ implies that of $M_1$ and by $M_0 \Rightarrow M_1$ that $M_0 \Rightarrow M_1$ and $M_1 \Rightarrow M_0$. We prove that $M_{c,x}^{\text{dist}} \Rightarrow M_{c,x}^{\text{DDH}}$ for any $X$, with $C \subseteq X$, $|C| = k$.

First, it is clear that $M_{c,x}^{\text{DDH}} \Rightarrow M_{c,x}^{\text{dist}}$ for any $X$, with $C \subseteq X$, $|C| = k$. Secondly, we show that $M_{c,x}^{\text{dist}} \Rightarrow M_{c,x}^{\text{DDH}}$ for any $X$, with $C \subseteq X$, $|C| = k$ by constructing $M_{c,x}^{\text{DDH}}$ using $M_{c,x}^{\text{dist}}$ as a subroutine. The construction of $M_{c,x}^{\text{DDH}}$ is as follows:

Algorithm 3 (Polynomial-time algorithm $M_{c,x}^{\text{DDH}}$)
Input: a challenge 4-tuple, $(g_1, g_2, g_3, g_4)$.
Output: “yes” or “no.”

Step 1. Choose a set of subscribers, $U \subseteq \mathbb{Z}_q \setminus \{0\}$, and split $U$ into $k$ disjoint subsets $U_1, \ldots, U_k$. Then, select a set of revoked subscribers, $X \subseteq U$, and a coalition of $k$ revoked subscribers, $C \subseteq X$.

Step 2. Suppose that $C = \{x_1, \ldots, x_k\}$ and $x_j \in U_i$, where $1 \leq j \leq k, i_j \in \{1, \ldots, k\}$. Choose $\lambda, \mu, a_1, \ldots, a_k, \beta_1, \ldots, \beta_k \in \mathbb{Z}_q$. Then, there exist two unique polynomials $\alpha(x) = \sum_{i=0}^k a_i x^i \mod q$. From Eq. (3) and Eq. (4), we can construct the public key $e$.

Note that no personal key other than those of the $k$ traitors cannot be computed from the values of $a_1, \ldots, a_k, \beta_1, \ldots, \beta_k$. From Eq. (3) and Eq. (4), we can construct the public key $e$.

Step 3. Check whether $Y = X \cup \cup_{j \in [k]} U_j$ is a non-empty set or not. If $Y = \{u_1, \ldots, u_m\}$, then find an integer $d$ s.t. $d(k+1) \leq w \leq d(k+1)+k$. Set $m = d(k+1)+k$.

Select $c_0, \ldots, c_m \in \mathbb{Z}_q$, and $u_1, \ldots, u_m \in \mathbb{Z}_q \setminus (\cup \cup_{j \in [k]} U_j \cup \{0\})$ if $w < m$. Then, build the header $h(r, X)$ as follows:

$$h(r, X) = (h_0, h_0, \ldots, h_0, h_{11}, \ldots, h_{1m}, H_1, \ldots, H_m),$$

where

$$h = g_3,$$

$$h_{0,j} = \begin{cases} g_1^{g_{1_k}^{\mu_j}} & (z_j = 0), \\ (g_1^{g_3^{\mu_j}})^{\mu_j} & (z_j \neq 0), \end{cases}$$

$$z_j = j \mod (k+1),$$

$$h_{1,j} = \begin{cases} g_1^{g_{1_k}^{\mu_j}} & (U_j \subseteq X, z_j \neq 0), \\ (g_1^{g_3^{\mu_j}})^{\mu_j} & (U_j \subseteq X, z_j \neq 0), \end{cases}$$

$$H_j = (u_j, g_{1_k}^{F(u_j)}),$$

$$F(x) = \sum_{\ell=0}^m c_\ell x^\ell \mod q,$$
and $r_j$ for each $j \in \{z \mid 1 \leq z \leq m, \ U_{z \mod (k+1)} \subseteq X\}$ is a random number. If $z_j = 0$, then $h_{1j}$ is not included in the header. Observe that if the challenge 4-tuple is a Diffie-Hellman tuple, $g_1^{\alpha_i}$ is the session key corresponding to $h(r, X)$. Otherwise, it does not correspond to this header.

Step 4. Give $g_1^\alpha, h(r, X), e, (x_1, i_1, d_1), \ldots, (x_k, i_k, d_k)$, and polynomially many previous session keys and headers to $M^\text{dist}_{C \subseteq X}$. If $M^\text{comp}_{C \subseteq X}$ decides that $g_1^\alpha$ is the session key corresponding to $h(r, X)$, then output "yes." Otherwise, output "no." Since $M^\text{comp}_{C \subseteq X}$ behaves differently for session keys and random elements in $G_q$, $M^\text{DDH}$ can solve the given DDH challenge.

Since $X, C$ with $C \subseteq X$, $|C| = k$ can be chosen arbitrarily in Step 1, it holds that $M^\text{comp}_{C \subseteq X} \Rightarrow M^\text{DDH}$ for any $X, C$ with $C \subseteq X$, $|C| = k$. This completes the proof. □

Lemma 1 leads to the next theorem which shows that, for any $X$, no coalition of at most $k$ revoked subscribers can compute the session key with non-negligible probability.

**Theorem 1** Suppose that the subscribers in $X$ are revoked after obtaining a certain number (bounded by a polynomial) of previous session keys and headers, the new header, the public key, and their personal keys. For any $X$, the computational complexity for any coalition of at most $k$ revoked subscribers to compute the session key corresponding to the new header is at least as difficult as DDH in $G_q$.

**Proof** We reuse the notations $C, M^\text{dist}_{C \subseteq X}, M^\text{DDH}$, and $\Leftrightarrow$ defined in the proof of Lemma 1. Let $M^\text{comp}_{C \subseteq X}$ be a polynomial-time algorithm the coalition $C$ uses to compute the session key corresponding to the new header when a set of revoked subscribers is $X$. Among $M^\text{comp}_{C \subseteq X}, M^\text{dist}_{C \subseteq X}$, and $M^\text{DDH}$ there exist the following three relations:

(R1) $M^\text{comp}_{C \subseteq X} \Rightarrow M^\text{dist}_{C \subseteq X}$ for any $X, C$ with $C \subseteq X$, $|C| = k$ (: $M^\text{dist}_{C \subseteq X}$ can be built via one oracle call to $M^\text{comp}_{C \subseteq X}$)

(R2) $M^\text{dist}_{C \subseteq X} \Leftrightarrow M^\text{DDH}$ for any $X, C$ with $C \subseteq X$, $|C| = k$ (: Lemma 1)

(R3) $M^\text{comp}_{C \subseteq X} \Rightarrow M^\text{comp}_{C' \subseteq X}$ for any $X, C$, and $C'$ with $C \subseteq X$, $|C| = k$, $C' \subseteq C$

It follows from (R1) and (R2) that $M^\text{comp}_{C \subseteq X} \Rightarrow M^\text{DDH}$ for any $X, C$ with $C \subseteq X$, $|C| = k$. From this result and (R3), it holds that $M^\text{comp}_{C \subseteq X} \Rightarrow M^\text{DDH}$ for any $X, C$ with $C \subseteq X$, $|C| \leq k$, which means the statement of Theorem 1. □

Note that we do not consider the collusion between revoked subscribers and non-revoked ones. Even if they collude, the system itself is not broken, i.e., they cannot compute the personal key of another subscriber, as long as the coalition size is not greater than $k$. In this case, the non-revoked ones can be identified as traitors and will be revoked. Therefore, the above assumption is reasonable when considering secrecy.

4.2 Black-Box Traceability

Let $C$ be a coalition of traitors and recall that $T_\ell$ is a set of suspects in the input for black-box tracing. Similarly to the previous schemes, the inputs for the normal broadcast and the ones for black-box tracing are indistinguishable. Therefore, the output of the pirate decoder on the input $h(r, T_\ell)$ should be:

- correct if $C \nsubseteq T_\ell$.
- incorrect with overwhelming probability if $C \subseteq T_\ell$ (=: Theorem 1).

**Lemma 2** If Algorithm 2 answers "correct" for $T_\ell$ and "incorrect" for $T_\ell \cup \{u\}$ ($u \notin T_\ell$), then the subscriber $u$ is a traitor with probability $1 - \varepsilon$ where $\varepsilon$ is negligible.

**Proof** Since Algorithm 2 outputs "incorrect" for $T_\ell \cup \{u\}$, it must hold that $C \subseteq T_\ell \cup \{u\}$ with overwhelming probability. Assume that $u \notin C$. Since $C \subseteq T_\ell$ in this case, Algorithm 2 must output "incorrect" for $T_\ell$. This is a contradiction. Hence, the case where $u \notin C$ is impossible. □

From Lemma 2, the following theorem holds.

**Theorem 2** From the pirate decoder constructed by a coalition of at most $k$ traitors, Algorithm 1 can identify at least one of them with probability $1 - \varepsilon$ where $\varepsilon$ is negligible.

**Proof** First, we prove that Algorithm 1 does not fail. Since Algorithm 2 must answer "correct" for $\emptyset$ and "incorrect" for $\mathcal{U}$, there exists two sets $T_0, T_1$ s.t. Algorithm 2 outputs "correct" for $T_0$ and "incorrect" for $T_1 \cup \{u\}$ ($u \notin T_1$), where $T_0 \subseteq T_1, |T_1 \setminus T_0| = 1$, $T_0 \subseteq \mathcal{U}$, and $T_2$ is any subset of $\mathcal{U} \setminus T_1$. It is clear that Algorithm 1 can find such sets $T_0, T_1$. Therefore, Algorithm 1 does not fail. Secondly, it immediately follows from Lemma 2 that the subscriber detected by Algorithm 1 is a traitor with overwhelming probability. This completes the proof. □

Note that the collusion attack [9], [16], which degrades the traceability of the scheme of [8] cannot be straightforwardly applied to our scheme since ours is based on not only the scheme of [8] but also that of [14] in which secret sharing in the exponents is performed. We also note that an elaborated version of the collusion attack is reported in [18] but this attack is successful only under a non-black-box model in which the tracer breaks open the pirate decoder and only checks if the keys embedded in it are the same as some personal keys. The attack is impossible under our black-box model since the result of black-box tracing is independent of a representation of the keys in the pirate decoder.
Table 1: Efficiency comparison (P, S, H): sets of possible personal keys, session keys, and headers respectively, n: the total number of subscribers, k: the maximum coalition size, t: the number of revoked subscribers, \( t_{\text{max}} \): the maximum number of revoked subscribers, d: an integer s.t. \( |Y| \leq d(k+1)+k \) where Y is a set of revoked subscribers who coexist with one or more non-revoked ones in each of their subsets.

|          | Each subscriber’s storage (\log |P|/\log |S|) | Transmission overhead (\log |H|/\log |S|) | Running time of the tracing algorithm | Is \( t_{\text{max}} \) variable in each distribution? | Computational cost for decryption |
|----------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|
| [11]     | 1                               | 2k + 1                          | \( O(k) \)                      | No (\( t_{\text{max}} = k \)) | \( O(k) \) exp.’s                 |
| [13]     | 1                               | 4k                              | \( O(k) \)                      | No (\( t_{\text{max}} = 2k - 1 \)) | \( O(k) \) exp.’s                 |
| [12]     | 1                               | 2(k + 1)                        | \( O(k) \)                      | No (\( t_{\text{max}} = k \)) | \( O(k) \) exp.’s                 |
| [7] (CS) | \( \log n + 1 \)               | \( \log n/\log |H| (\text{worst case}) \) | \( O(\log n) \)                 | Yes                             | \( 1 \) comp. of a bilinear map   |
| Ours     | 1                               | \( 4(d + 1)k + 3d + 2 \)        | \( O(\log n) \)                 | Yes                             | \( O(d + 1)k \) exp.’s            |

5. Efficiency

In Table 1, the previous schemes and ours are compared from the viewpoints of each subscriber’s storage, the transmission overhead, the running time of the tracing algorithm, and the type of revocation. In our scheme, each subscriber’s storage is constant and the transmission overhead does not increase substantially over the previous schemes. If a set of subscribers \( U \) can be split into \( U_1, \ldots, U_k \) so that it can be a usual case that bulk revocation occurs, where we mean by bulk revocation that most of the revoked subscribers belong to the subgroups in each of which all of the subscribers are revoked, then the transmission overhead is linear only in \( k \). For example, it holds that \( \log |H|/\log |S| = 4k + 2 \) when \( d = 0 \), i.e., the number of revoked subscribers who are non-bulk-revoked ones is not greater than \( k \). In our scheme, the data supplier can make it impossible for the revoked subscribers to compute the session key by (i) substituting a random value for the element used only by the subscribers in one of the \( k \) disjoint subsets if all of them in the subset are revoked or (ii) adding their shares of the session key to the header. By using the former method, the corresponding shares of the session key can be unnecessary in the header, while all of the revoked-subscribers’ shares are required to be added to the header in the schemes of [11]–[13] since they support the latter method only. Revocation in our scheme is flexible in the sense that the maximum number of revoked subscribers does not have to be fixed in the key-generation phase and is variable in each distribution, although the security level of revocation remains constant. It is desirable that the capacity level of revocation should be variable in each distribution in the case where the range of the number of revoked subscribers is wide.

The outstanding advantage of our scheme is the efficient running time of the tracing algorithm. In our scheme, at least one of the traitors in a coalition can be caught with running time \( O(\log n) \), while the running time of the previous one is \( O\left(\binom{|Y|}{t}\right) \). Thanks to flexible revocation, an efficient tracing algorithm can be achieved without a substantial increase in the transmission overhead. Note that, due to lack of the flexibility in revocation, the scheme of [12] has to incur an inefficient transmission overhead, which is linear in \( n \), in order to achieve efficient black-box tracing with running time \( O(\log n) \). This problem is common to the schemes of [11], [13].

The scheme of [7]† makes use of the personal-key assignment based on a binary tree and identity-based encryption [19]. Similar to our scheme, the scheme achieves efficient running time of the tracing algorithm and can reduce the transmission overhead in case of bulk revocation. The main difference is that each subscriber’s storage and the transmission overhead depend on \( n \) in the previous scheme, while in our scheme they do not. This difference is significant when new subscribers subsequently join the system. In order to deal with the late subscription, a sufficiently large binary-tree must be supposed when the system is initialized. This capacity margin results in an increase in each subscriber’s storage and the transmission overhead. In our scheme, there is no such problem. Regarding traceability, in the previous scheme it needs to be assumed that the pirate decoder has all of the node keys in the path from at least one traitor’s leaf to the root, i.e., at least one set of personal keys. For example, if the traitor simply embeds all of his node keys other than his leaf key to the pirate decoder, it can decrypt most of the headers and the tracing algorithm cannot determine whether a traitor is him or not with overwhelming probability, although all of the embedded keys can be revoked. In our scheme, no such assumption is needed. On the other hand, the computational cost for decryption remains to be improved in our scheme. The number of exponentiations required for decryption is \( O(d + 1)k \) in ours, while in the previous scheme a single identity-based decryption is needed. The cost for decryption in our scheme can be reduced thanks to vector-addition chain exponentiation [20, p.622], although there is a trade-off between the decryption complexity and the maximum coalition size.

6. Conclusions

In this paper, we have proposed a flexible-revocation scheme for efficient public-key black-box traitor tracing. Our scheme is efficient in all of the following criteria: each subscriber’s storage, the transmission overhead, and the running time of the tracing algorithm. Thanks to flexible revocation, which makes it possible to change the maximum number of revoked subscribers in each distribution, the efficient tracing algorithm can be applied to our scheme without a substantial increase in the transmission overhead.

†We suppose a public-key extension of the Complete Subtree method [5] since it is more efficient than that of the Subset Difference method [5] as analyzed in [7].
Acknowledgment

We would like to thank the anonymous reviewers for their helpful comments.

References


Tatsuyuki Matsushita received the B.E. degree in electrical and electronic engineering from Kyoto University in 1999, and the M.E. degree in information and communication engineering from the University of Tokyo in 2001. He is a researcher of TOSHIBA Corporate Research & Development Center and a Ph.D. student of the University of Tokyo.

Hideki Imai was born in Shimane, Japan on May 31, 1943. He received the B.E., M.E., and Ph.D. degrees in electrical engineering from the University of Tokyo in 1966, 1968, 1971, respectively. From 1971 to 1992 he was on the faculty of Yokohama National University. In 1992 he joined the faculty of the University of Tokyo, where he is currently a Full Professor in the Institute of Industrial Science. His current research interests include information theory, coding theory, cryptography, spread spectrum systems and their applications. From IEICE (the Institute of Electronics, Information and Communication Engineers) he received Best Book Awards in 1976 and 1991, Best Paper Awards in 1992, 2003 and 2004, Yonezawa Memorial Paper Award in 1992, Achievement Award in 1995, Inose Award in 2003, and Distinguished Achievement and Contributions Award in 2004. He also received Golden Jubilee Paper Award from the IEEE Information Theory Society in 1998, and official Commendations from the Minister of Public Management, Home Affairs, Posts and Telecommunications in June 2002 and from the Minister of Economy, Trade and Industry in October 2002. He was awarded Honor Doctor Degree by Soochunhyang University, Korea in 1999 and Doctor Honoris Causa by the University of Toulon Var, France in 2002. He was elected an IEEE Fellow in 1992 and an IEICE Fellow in 2001. He chaired several committees of scientific societies and organized many international conferences such as IEEE-ITW, IEEE-ISIT, AAEC, PKC, FSE, and WPMC. He served as the leader of research projects supported by JSPS (Japan Society for the Promotion of Science), IPA (Information-technology Promotion Agency, Japan) etc. and as the editor for scientific journals of IEICE, IEEE etc. Dr. Imai was on the board of IEICE (1992–1994, 1996–1999), the IEEE Information Theory Society (IT-SOC, 1993–1998), Japan Society of Security Management (1988–present) and the Society of Information Theory and Its Applications (SITA, 1981–1997). He served as the president of SITA (1997), IEICE Engineering Sciences Society (1998–1999), IEEE Information Theory Society (2004–present), and as the chairman of CRYPTREC (Cryptography Techniques Research and Evaluation Committee of Japan) (2000–present).