UNIVERSITÀ DEGLI STUDI DI MILANO
FACOLTÀ DI SCIENZE E TECNOLOGIE
DIPARTIMENTO DI INFORMATICA

Traitor Tracing Schemes for Digital Content Protection

Chiara Valentina Schiavo
chiara.schiavo@unimi.it

BunnyTN4 - May 22nd, 2013

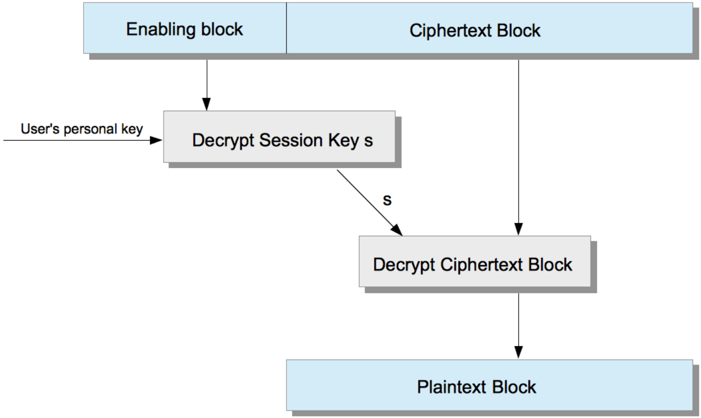
Outline

Outline

Context:

- Digital content distribution systems
- Authorized users are given a hardware or software decoder containing a decryption key that allows them to get access to the content in clear.
- The **content provider** broadcasts the encrypted content
- The **subscribers** (i.e., authorized users who pay for the service) use their own secret key to decrypt the digital content

How does it work?



Piracy

Problem

- The **traitors** (i.e., malicious subscribers) may collude and try to use their personal keys to construct a pirate decoder, i.e., a non-registered decoder able to decrypt
- Using the pirate decoder the **pirates** (i.e., unauthorized users) can illegally decrypt the digital contents

Piracy

Problem

- The **traitors** (i.e., malicious subscribers) may collude and try to use their personal keys to construct a pirate decoder, i.e., a non-registered decoder able to decrypt
- Using the pirate decoder the **pirates** (i.e., unauthorized users) can illegally decrypt the digital contents

Solution

Traitor Tracing Schemes: designed with the aim of identify (at least one of) the traitors, after the pirate decoder is confiscated

Traitor Tracing Scheme

A traitor tracing scheme is composed of four phases:

- *Key Generation*: the data supplier 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. The data supplier broadcasts the encrypted digital contents and the header.
- *Decryption*: subscribers retrieve the session key by inputting the header into their decoders.
- *Tracing*: after a pirate decoder confiscation, the tracer builds ad-hoc header in which suspected are revoked and uses the decoder as a black box.

Outline

First public-key tracing scheme for tracing illicit decoders that may shut-down (or employ some sort of self-defensive mechanism) (AsiaCrypt 2004).

Parameters

- n: total number of subscribers
- k: maximum number of traitors in a coalition
- p, q: primes s.t. $q|p-1$, $q \geq n+2k-1$
- g: a q-th root of unity over $\mathbb{Z}_p^*$
- $\mathcal{U}$: set of subscribers

Participants agree on p, q and g

Matsushita - Imai's Protocol

Key Generation

- Split $\mathcal{U}$ into ℓ disjoint subsets $\mathcal{U}_0, \ldots, \mathcal{U}_{\ell-1}$.
- Choose $a_0, \ldots, a_{2k-1}, b_0, \ldots, b_{\ell-1} \in_{\mathbb{R}} \mathbb{Z}_q$.
- Assign a distinct key-generation polynomial to each subset:

$$\begin{array}{l}
\mathcal{U}_0 \longleftarrow f_0(x) = b_0 + a_1 x + a_2 x^2 + \cdots + a_{2k-1} x^{2k-1} \mod q \\
\mathcal{U}_1 \longleftarrow f_1(x) = a_0 + b_1 x + a_2 x^2 + \cdots + a_{2k-1} x^{2k-1} \mod q \\
\ldots \\
\ldots \\
\mathcal{U}_i \longleftarrow f_i(x) = a_0 + a_1 x + \cdots + b_i x^i + \cdots + a_{2k-1} x^{2k-1} \mod q \\
\ldots
\end{array}$$

- Personal key of the user $u \in \mathcal{U}_i$:

$$(u, i, f_i(u))$$

- Public key:

$$\begin{aligned}
e &= (g, y_{0,0}, \ldots, y_{0,2k-1}, y_{1,0}, \ldots, y_{1,\ell-1}) \\
&= (g, g^{a_0}, \ldots, g^{a_{2k-1}}, g^{b_0}, \ldots, g^{b_{\ell-1}})
\end{aligned}$$

Encryption

- Select the session key $s \in G_q$ and two random numbers $R_0, R_1 \in_R \mathbb{Z}_q$.
- Choose $r_i \in \{R_0, R_1\}$ and construct the header H_i for the subgroup $\mathcal{U}_i$:

$$H_i = (\hat{h}_i, h_{i,0}, \ldots, h_{i,i}, \ldots, h_{i,2k-1}) = (g^{r_i}, y_{0,0}^{r_i}, y_{0,1}^{r_i}, \ldots, \mathbf{s}y_{1,i}^{r_i}, \ldots, y_{0,2k-1}^{r_i})$$
$$= (g^{r_i}, g^{a_0 r_i}, g^{a_1 r_i}, \ldots, \mathbf{sg^{b_i r_i}}, \ldots, g^{a_{2k-1} r_i})$$

- The data provider broadcasts the encrypted contents and the header $H = \{H_0, \ldots, H_{\ell-1}\}$

Matsushita - Imai's Protocol

Encryption

- Select the session key $s \in G_q$ and two random numbers $R_0, R_1 \in_R \mathbb{Z}_q$.
- Choose $r_i \in \{R_0, R_1\}$ and construct the header H_i for the subgroup $\mathcal{U}_i$:

$$H_i = (\hat{h}_i, h_{i,0}, \ldots, h_{i,i}, \ldots, h_{i,2k-1}) = (g^{r_i}, y_{0,0}^{r_i}, y_{0,1}^{r_i}, \ldots, \mathbf{s}y_{1,i}^{r_i}, \ldots, y_{0,2k-1}^{r_i})$$
$$= (g^{r_i}, g^{a_0 r_i}, g^{a_1 r_i}, \ldots, \mathbf{sg^{b_i r_i}}, \ldots, g^{a_{2k-1} r_i})$$

- The data provider broadcasts the encrypted contents and the header $H = \{H_0, \ldots, H_{\ell-1}\}$

Revocation

Users in $\mathcal{U}_i$ can be revoked by replacing $sg^{b_i r_i}$ with g^{z_i}, $(z_i \in_R \mathbb{Z}_q)$:

$$H_i = (g^{r_i}, g^{a_0 r_i}, g^{a_1 r_i}, \ldots, \mathbf{g^{z_i}}, \ldots, g^{a_{2k-1} r_i})$$

Matsushita - Imai's Protocol

Decryption

User $u \in \mathcal{U}_i$ computes the session key s from $H_i = (\hat{h}_i, h_{i,0}, \ldots, h_{i,i}, \ldots, h_{i,2k-1})$:

$$\left\{ \frac{\left(h_{i,0} \times (h_{i,1})^{u^1} \times \cdots \times (h_{i,2k-1})^{u^{2k-1}} \right)}{\hat{h}_i^{f_i(u)}} \right\}^{1/u^{i} \bmod 2k} = s$$

Matsushita - Imai's Protocol

Decryption

User $u \in \mathcal{U}_i$ computes the session key s from $H_i = (\hat{h}_i, h_{i,0}, \ldots, h_{i,i}, \ldots, h_{i,2k-1})$:

$$\left\{ \frac{\left(h_{i,0} \times (h_{i,1})^{u^1} \times \cdots \times (h_{i,2k-1})^{u^{2k-1}} \right)}{\hat{h}_i^{f_i(u)}} \right\}^{1/u^i \bmod 2k} = s$$

Black-box Tracing

Goal: *Identify at least one traitor.*

- Input: $\mathcal{U}_0, \ldots, \mathcal{U}_{\ell-1}$ and the pirate decoder (we assume $|\mathcal{U}_0| = \cdots = |\mathcal{U}_{\ell-1}| = 2k$)
- Output: Traitor identity
- For each user u_j with $1 \leq j \leq n$, set $ctr_j = 0$ and repeat m times the *Black-Box Tracing Test*. In each test s, R_0, R_1 are randomly chosen.
- Find an integer $j \in \{1, \ldots, n\}$ s.t. $ctr_{j-1} - ctr_j$ is maximum. The subscriber u_j is a traitor.

Outline

1. Introduction to the traitor tracings
2. An example: the Matsushita - Imai tracing scheme
3. A possible attack on the scheme
4. A way to totally repair the scheme

Encryption

The header H_i for the subgroup $\mathcal{U}_i$ is:

$$H_i = (\hat{h}_i, h_{i,0}, \ldots, h_{i,i}, \ldots, h_{i,2k-1}) = (g^{r_i}, g^{a_0 r_i}, g^{a_1 r_i}, \ldots, sg^{b_i r_i}, \ldots, g^{a_{2k-1} r_i})$$

with $\mathbf{r_i} \in \{\mathbf{R_0}, \mathbf{R_1}\}$ uniformly at random

Black-Box Tracing

$\mathcal{X} = \{u_1, \ldots, u_j\}$: set of revoked subscribers.
If there exists $\mathcal{U}_t$ such that $\mathcal{X} \cap \mathcal{U}_t \neq \emptyset$ and $\mathcal{X} \cap \mathcal{U}_t \neq \mathcal{U}_t$, then users in $\mathcal{U}_i$ with $i > t$ will receive H_i computed as follows:

$$H_i = (\hat{h}_i, h_{i,0}, \ldots, h_{i,i}, \ldots, h_{i,2k-1}) = (g^{r_i}, g^{a_0 r_i}, g^{a_1 r_i}, \ldots, sg^{b_i r_i}, \ldots, g^{a_{2k-1} r_i})$$

with $\mathbf{r_i} = \mathbf{R_0}$

Attack 1 [Kiayias et.al.]

Distinguish Normal Ciphertext from Tracing Ciphertext

r_i distribution in Normal Ciphertext case. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
$0/1$	...	$0/1$	$0/1$	**$0/1$**	...	**$0/1$**

r_i distribution in Tracing Ciphertext case. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
$0/1$	...	$0/1$	1	**0**	...	**0**

Attack 1 [Kiayias et.al.]

Distinguish Normal Ciphertext from Tracing Ciphertext

r_i distribution in Normal Ciphertext case. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	0/1	**0/1**	...	**0/1**

r_i distribution in Tracing Ciphertext case. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	1	**0**	...	**0**

- The pirate decoder can distinguish between tracing and regular system operations:

$$Adv_{decoder} = |P_{C \leftarrow Enc}[D(C) = 1] - P_{C \leftarrow Trace}[D(C) = 1]| =$$
$$= 1 - 2^{-k} - negl$$

where $k = |\{i | \mathcal{U}_i \cap T \neq 0\}|$ and *negl* is a negligible probability.

Attack 1 [Kiayias et.al.]

Distinguish Normal Ciphertext from Tracing Ciphertext

r_i distribution in Normal Ciphertext case. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	0/1	**0/1**	...	**0/1**

r_i distribution in Tracing Ciphertext case. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	1	**0**	...	**0**

- The pirate decoder can distinguish between tracing and regular system operations:

$$Adv_{decoder} = |P_{C \leftarrow Enc}[D(C) = 1] - P_{C \leftarrow Trace}[D(C) = 1]| =$$
$$= 1 - 2^{-k} - negl$$

  where $k = |\{i | \mathcal{U}_i \cap T \neq 0\}|$ and *negl* is a negligible probability.
- The pirate decoder can launch a self-defensive mechanism and accuse an innocent user.

Attack 2 [Kiayias et.al.]

Gap between $CTrace(e, j-1, s)$ and $CTrace(e, j, s)$

- $j \equiv 1 \mod 2k$

r_i distribution in case $CTrace(e, j-1, \cdot)$. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	0/1	**0/1**	...	**0/1**

r_i distribution in case $CTrace(e, j, \cdot)$. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	1	**0**	...	**0**

Attack 2 [Kiayias et.al.]

Gap between $CTrace(e, j-1, s)$ and $CTrace(e, j, s)$

- $j \equiv 1 \mod 2k$

r_i distribution in case $CTrace(e, j-1, \cdot)$. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	0/1	**0/1**	...	**0/1**

r_i distribution in case $CTrace(e, j, \cdot)$. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	1	**0**	...	**0**

- The pirate decoder can distinguish the gap and launch a self-defensive mechanism to accuse an innocent user

Kiayias - Pehlivanoglu Solution

Encryption Phase Modification:

- Select a random cutoff point $d \in \{0, \ldots, \ell - 1\}$.
  Set $r_i = R_1$ for $i \leq d$ and $r_i = R_0$ for $i > d$

Distribution of r_i. ($R_0 = 0$ and $R_1 = 1$)

	$\mathcal{U}_0$	$\ldots$	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	$\ldots$	$\mathcal{U}_{\ell-1}$
Original Encryption	$0/1$	$\ldots$	$0/1$	$0/1$	**$0/1$**	$\ldots$	**$0/1$**

	$\mathcal{U}_0$	$\ldots$	$\mathcal{U}_{d-1}$	$\mathcal{U}_d$	$\mathcal{U}_{d+1}$	$\ldots$	$\mathcal{U}_{\ell-1}$
Modified Encryption	1	$\ldots$	1	1	**0**	$\ldots$	**0**

	$\mathcal{U}_0$	$\ldots$	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	$\ldots$	$\mathcal{U}_{\ell-1}$
Tracing Encryption	$0/1$	$\ldots$	$0/1$	1	**0**	$\ldots$	**0**

Limitations

- the scheme is **still susceptible to the Attack 1**, depending on d and t.
- the proposed solution **does not fix the Attack 2**: the statistical gap between $CTrace(e, j-1, s)$ and $CTrace(e, j, s)$ still remains.
- the **pirate decoder can still avoid the tracing phase**
- **an innocent user is accused**

Outline

1. Introduction to the traitor tracings
2. An example: the Matsushita - Imai tracing scheme
3. A possible attack on the scheme
4. A way to totally repair the scheme

Goal

Completely repair the Matsushita and Imai's traitor tracing scheme:

- prevent the pirate decoder from recognizing normal ciphertexts from tracing ciphertexts (Attack 1)
- close the statistical distance between two consecutive tracing ciphertexts (Attack 2)

Repairing the scheme

Goal

Completely repair the Matsushita and Imai's traitor tracing scheme:

- prevent the pirate decoder from recognizing normal ciphertexts from tracing ciphertexts (Attack 1)
- close the statistical distance between two consecutive tracing ciphertexts (Attack 2)

In this way:

- the pirate decoder can not evade the tracing activity
- no innocent user is incriminated
- at least one of the traitors is identified

Repairing the scheme

What we have...

r_i distribution: **Normal Ciphertext** case. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	0/1	**0/1**	...	**0/1**

r_i distribution: **Tracing Ciphertext** case. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	1	**0**	...	**0**

What we'd like to have....

r_i distribution: **Normal Ciphertext** case. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	0/1	**0/1**	...	**0/1**

r_i distribution: **Tracing Ciphertext** case. ($R_0 = 0$ and $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	1	**0/1**	...	**0/1**

Repairing the scheme

What we have...

r_i distribution: $\mathbf{CTrace}(\mathbf{e}, \mathbf{j-1}, \cdot)$. ($R_0 = 0$ e $R_1 = 1$)

$\mathcal{U}_0$	$\dots$	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	$\dots$	$\mathcal{U}_{\ell-1}$
$0/1$	$\dots$	$0/1$	$0/1$	$\mathbf{0/1}$	$\dots$	$\mathbf{0/1}$

r_i distribution: $\mathbf{CTrace}(\mathbf{e}, \mathbf{j}, \cdot)$. ($R_0 = 0$ e $R_1 = 1$)

$\mathcal{U}_0$	$\dots$	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	$\dots$	$\mathcal{U}_{\ell-1}$
$0/1$	$\dots$	$0/1$	1	$\mathbf{0}$	$\dots$	$\mathbf{0}$

What we'd like to have....

r_i distribution: $\mathbf{CTrace}(\mathbf{e}, \mathbf{j-1}, \cdot)$. ($R_0 = 0$ e $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	0/1	**0/1**	...	**0/1**

r_i distribution: $\mathbf{CTrace}(\mathbf{e}, \mathbf{j}, \cdot)$. ($R_0 = 0$ e $R_1 = 1$)

$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
0/1	...	0/1	1	**0/1**	...	**0/1**

Requirement

Indistinguishability of an input: revoked users should not be able to distinguish tracing and regular system operation

Warning

The headers H_i can not be constructed as in the original protocol.

Solution

- Redesign the tracing phase in order to:
  - close the statistical gap between the normal ciphertext and the tracing ciphertext
  - close the statistical gap between two consecutive tracing ciphertext
- Modify the header construction procedure to allow the correct decryption (i.e. to retrieve the session key)
- Update Over The Air

Conclusions

Theorem

Given the new traitor tracing scheme and a pirate decoder constructed by a coalition of k traitors, at least one of the traitors can be identified with probability $1 - \epsilon$, where ϵ is negligible.

Conclusions

Theorem

Given the new traitor tracing scheme and a pirate decoder constructed by a coalition of k traitors, at least one of the traitors can be identified with probability $1 - \epsilon$, where ϵ is negligible.

The Matsushita-Imai traitor tracing scheme is completely repaired
The new Traitor Tracing Scheme ensures that:

- at least one traitor is identified
- the pirate decoder is not able to recognize normal ciphertext from tracing ciphertext (resistant to Attack 1)
- the pirate decoder is not able to recognize two consecutive tracing operations (resistant to Attack 2)
- the pirate decoder can not avoid the tracing activity
- no innocent user is illegitimately accused

Thank you for your attention!

Chiara Valentina Schiavo
chiara.schiavo@unimi.it

Backup slides

The original Black-Box Tracing phase

Black-box Tracing Test

1. $\mathcal{X} = \{u_1, \ldots, u_j\}$: set of revoked subscribers. Construct the header $H = (H_0, \ldots, H_{\ell-1})$ where each H_i is as follows:
   - if $\mathcal{X} \cap \mathcal{U}_i = \mathcal{U}_i$ or $\mathcal{X} \cap \mathcal{U}_i = \emptyset$ for any i, then the header H_i is constructed as in the encryption phase, with the revoking value g^{z_i} when $\mathcal{X} \cap \mathcal{U}_i = \mathcal{U}_i$
   - Otherwise, if there exists $\mathcal{U}_t$ such that $\mathcal{X} \cap \mathcal{U}_t \neq \emptyset$ and $\mathcal{X} \cap \mathcal{U}_t \neq \mathcal{U}_t$ then construct $C(x) = \sum_{j=0}^{2k-1} c_j x^j$ s.t. $C(u) = 0 \mod q$ iif $u \in (\mathcal{U}_t \backslash \mathcal{X})$. Then H_i is constructed as:
     - if $i = t$ then H_i is as follows:

$$\hat{h}_t = g^{R_1}$$

$$h_{t,j} = \begin{cases} g^{c_j} y_{0,j}^{R_1} & j \neq t \bmod 2k \\ s g^{c_j} y_{1,t}^{R_1} & j = t \bmod 2k \end{cases}$$

     - if $i > t$ then H_i is computed as in the encryption phase with $r_i = R_0$.
     - if $i < t$ then $r_i \in \{R_0, R_1\}$ random. H_i is computed as follows:

$$\hat{h}_i = g^{r_i}, \qquad r_i = R_0 \text{ or } R_1$$

$$h_{i,j} = \begin{cases} y_{0,j}^{R_0} & j \neq i \bmod 2k,\ r_i = R_0 \\ g^{c_j} y_{0,j}^{R_1} & j \neq i \bmod 2k,\ r_i = R_1 \\ g^{z_i} & j = i \bmod 2k \end{cases}$$

2. Give H to the pirate decoder and monitor the output
3. If the pirate decoder decrypts correctly, then increment ctr_j by 1.

The new Black-Box Tracing phase

Original Tracing Phase

1. $\mathcal{X} = \{u_1, \ldots, u_j\}$: set of revoked subscribers. Construct the header $H = (H_0, \ldots, H_{\ell-1})$ where each H_i is as follows:
   - If there exists $\mathcal{U}_t$ such that $\mathcal{X} \cap \mathcal{U}_t \neq \emptyset$ and $\mathcal{X} \cap \mathcal{U}_t \neq \mathcal{U}_t$, then ...
     - if $i = t \ldots$
     - **if $i > t$, then H_i is computed as in the encryption phase with $r_i = R_0$**
     - if $i < t \ldots$
   - otherwise . . .
2. Give H to the pirate decoder and monitor the output
3. If the decoder decrypts correctly, increment ctr_j by 1.

Our Modified Tracing Phase

1. $\mathcal{X} = \{u_1, \ldots, u_j\}$: set of revoked subscribers. Construct the header $H = (H_0, \ldots, H_{\ell-1})$ where each H_i is as follows:
   - If there exists $\mathcal{U}_t$ such that $\mathcal{X} \cap \mathcal{U}_t \neq \emptyset$ and $\mathcal{X} \cap \mathcal{U}_t \neq \mathcal{U}_t$, then ...
     - if $i = t \ldots$
     - **if $i \neq t$, then $r_i = R_0$ or $r_i = R_1$. Construct H_i as follows ....**
   - otherwise . . .
2. Give H to the pirate decoder and monitor the output
3. If the decoder decrypts correctly, increment ctr_j by 1.

Solution: new Black-Box Tracing phase

Our Modified Black-box Tracing

1. $\mathcal{X} = \{u_1, \ldots, u_j\}$: set of revoked subscribers. Construct the header $H = (H_0, \ldots, H_{\ell-1})$ where each H_i is as follows:
   - If there exists $\mathcal{U}_t$ such that $\mathcal{X} \cap \mathcal{U}_t \neq \emptyset$ and $\mathcal{X} \cap \mathcal{U}_t \neq \mathcal{U}_t$, then ...
     - if $i = t$, then H_i is computed as in the original protocol
     - **if $i \neq t$, then $r_i = R_0$ or $r_i = R_1$. Construct H_i as follows:**

$$\hat{h}_i = g^{r_i}, \qquad r_i = R_0 \text{ or } R_1$$

$$h_{i,j} = \begin{cases} y_{0,j}^{R_0} & j \neq i \bmod 2k,\ r_i = R_0 \\ g^{c_j} y_{0,j}^{R_1} & j \neq i \bmod 2k,\ r_i = R_1 \\ \mathbf{s y_{1,i}^{R_0}} & \mathbf{j = i \bmod 2k,\ i > t,\ r_i = R_0} \\ \mathbf{s g^{c_j} y_{1,i}^{R_1}} & \mathbf{j = i \bmod 2k,\ i > t,\ r_i = R_1} \\ g^{z_i} & j = i \bmod 2k,\ i < t \end{cases}$$

   - otherwise if $\mathcal{X} \cap \mathcal{U}_i = \emptyset$ or $\mathcal{X} \cap \mathcal{U}_i = \mathcal{U}_i$ for any i, then the header H_i is the same as in the encryption phase setting the revocation parameters.
2. Give H to the pirate decoder and monitor the output
3. If the pirate decoder decrypts correctly, then increment ctr_j by 1.

Decryption in case $i > t$ and $r_i = R_1$

The header is $H_i = (\hat{h}_i, h_{i,0}, \dots, h_{i,i}, \dots, h_{i,2k-1})$.

The secret session key s is retrieved as follows:

$$\left\{ \frac{h_{i,0} \times h_{i,1}^{u} \times \cdots \times h_{i,2k-1}^{u^{2k-1}}}{\hat{h}_i^{f_i(u)}} \right\}^{1/u^i \bmod 2k}$$

$$= \left\{ \frac{g^{c_0} y_{0,0}^{R_1} \times \cdots \times \left(s g^{c_j} y_{1,j}^{R_1} \right)^{u^i \bmod 2k} \times \cdots \times \left(g^{c_{2k-1}} y_{0,2k-1}^{R_1} \right)^{u^{2k-1}}}{g^{R_1 f_i(u)}} \right\}^{1/u^i \bmod 2k}$$

$$= \left\{ \frac{g^{c_0} g^{a_0 R_1} \times \cdots \times \left(s g^{c_i} g^{b_i R_1} \right)^{u^i \bmod 2k} \times \cdots \times \left(g^{c_{2k-1}} g^{a_{2k-1} R_1} \right)^{u^{2k-1}}}{g^{R_1 f_i(u)}} \right\}^{1/u^i \bmod 2k}$$

$$= \left\{ \frac{s^{u^i \bmod 2k} g^{\sum_{j=0}^{2k-1} c_j u^j} g^{R_1 (\sum_{j=0}^{2k-1} a_j u^j + b_i u^i - a_i u^i)}}{g^{R_1 f_i(u)}} \right\}^{1/u^i \bmod 2k} = s$$

Our Results

Distribution of r_i with Normal Ciphertext and Tracing Ciphertext

	$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
M-I scheme - Normal ciphertext	R_0/R_1	...	R_0/R_1	R_0/R_1	R_0/R_1	...	R_0/R_1
M-I scheme - Tracing ciphertext	R_0/R_1	...	R_0/R_1	R_1	R_0	...	R_0
Our scheme - Normal ciphertext	R_0/R_1	...	R_0/R_1	R_0/R_1	R_0/R_1	...	R_0/R_1
Our scheme - Tracing ciphertext	R_0/R_1	...	R_0/R_1	R_1	R_0/R_1	...	R_0/R_1

Distribution of r_i, case $CTrace(e, j-1, \cdot)$ and $CTrace(e, j, \cdot)$

	$\mathcal{U}_0$	...	$\mathcal{U}_{t-1}$	$\mathcal{U}_t$	$\mathcal{U}_{t+1}$	...	$\mathcal{U}_{\ell-1}$
M-I scheme: $CTrace(e, j-1, \cdot)$	R_0/R_1	...	R_0/R_1	R_0/R_1	R_0/R_1	...	R_0/R_1
M-I scheme: $CTrace(e, j, \cdot)$	R_0/R_1	...	R_0/R_1	R_1	R_0	...	R_0
Our scheme: $CTrace(e, j-1, \cdot)$	R_0/R_1	...	R_0/R_1	R_0/R_1	R_0/R_1	...	R_0/R_1
Our scheme: $CTrace(e, j, \cdot)$	R_0/R_1	...	R_0/R_1	R_1	R_0/R_1	...	R_0/R_1