Sequential Traitor Tracing

Reihaneh Safavi-Naini and Yejing Wang

School of IT and CS, University of Wollongong,
Wollongong 2522, Australia
[rei/yw17]@uow.edu.au

Abstract. Traceability schemes allow detection of at least one traitor when a group of colluders attempt to construct a pirate decoder and gain illegal access to digital content. Fiat and Tassa proposed dynamic traitor tracing schemes that can detect *all* traitors if they attempt to re-broadcast the content after it is decrypted. In their scheme the content is broken into segments and marked so that a re-broadcasted segment can be linked to a particular subgroup of users. Mark allocation for a segment is determined when the re-broadcast from the previous segment is observed. They showed that by careful design of the mark allocation scheme it is possible to detect all traitors.

We consider the same scenario as Fiat and Tassa and propose a new type of traceability scheme, called sequential traitor tracing, that can efficiently detect *all* traitors and does not require any real-time computation. That is, the marking allocation is pre-determined and is independent of the re-broadcasted segment. This is very attractive as it allows segments to be shortened and hence the overall convergence time reduced. We analyse the scheme and give two general constructions one based on a special type of function family, and the other on error correcting codes. We obtain the convergence time of these schemes and show that the scheme based on error correcting codes has a convergence time which is the same as the best known result for dynamic schemes.

1 Introduction

In recent years a number of closely related models and schemes with the aim of securing electronic distribution of digital content have been proposed. Services that rely on this kind of distribution, such as pay-TV where the provider needs assurance that only paid customers will receive the service, can only become viable if security of the distribution can be guaranteed.

Broadcast encryption systems [6] allow targeting of an encrypted message to a privileged group of receivers. Each receiver has a decoder with his unique key information that allows him to decrypt encrypted messages when he belongs to the target group. The system ensures that the collusion of up to t receivers not belonging to the target group cannot learn anything about the message.

M. Bellare (Ed.): CRYPTO 2000, LNCS 1880, pp. 316–332, 2000.

Now assume a group of colluders construct a *pirate decoder* that can decrypt the broadcasted message. A *traitor tracing scheme* [4] allows at least one of the colluders to be identified. Broadcast encryption and traceability systems can be combined [8,10] to produce systems that provide protection against both kinds of attacks.

In [7] a different scenario is considered. This time the traitors' aim is to bypass the security mechanism of the system not by constructing a pirate decoder but by *re-broadcasting the content* after it is decoded. That is the colluders use their decoders to decrypt the content and then once it is in plain-text form, re-broadcast the plain-text to another group of users. In this case the only way to trace traitors is to use different *versions* of the content for different users. This allows a re-broadcasted message to be linked to the subgroup of users who were given that version. The two main characteristics of the new setting are (i) the plaintext content is marked and, (ii) there is a *feedback* from the channel which allows the traitor to become localized. An important feature of this system is that it allows *all* traitors, even without knowing their number beforehand, be traced.

A trivial solution to tracing traitors in the above scenario is to give individual copies to each users. This means that the same content must be sent once per each user and so bandwidth usage is extremely poor. Fiat and Tassa's (FT for short) work showed that by introducing a new dimension to the problem, that is *time*, it is possible to use a small number of versions and hence resulting in a more efficient usage of the communication channel, while still detecting all the traitors. In FT system the content is divided into *consecutive segments* and each segment is *watermarked* to produce ℓ versions. For each segment users are partitioned into ℓ subgroups, and members of a subgroup receive the same marked version. The system is *dynamic* because the version received by a user, and hence partitioning of the user set, depends on the feedback from previous segment. The number of traitors is not known beforehand and the system adjusts itself such that as long as the total number of traitors is less than p, all of them will be traced. The algorithms proposed in [7] allow trade-off between the two efficiency parameters of these systems, that is, bandwidth usage and convergence time.

Drawbacks of FT model: There are two major drawbacks in FT model. Firstly, mark allocation in a time slot depends on the *real-time feedback* signal from the previous time slot. This makes the system vulnerable to *delayed rebroadcast attack*. That is, when the attackers do not rebroadcast immediately, but decide to record the content and rebroadcast it at a later time. In this case FT model becomes totally ineffective as the mark allocation in time slots will remain constant.

The second drawback is the high real-time computation required for allocation of marks which means the length of a time-slot cannot be very short. We note that the number of time slots for the convergence of the best proposed algorithm is at least of the order of $\log N$ (N is the number of users) and hence grows with the number of users. For large group sizes, it is desirable to have shorter time slots to obtain reasonable convergence time. However the computation grows with the size of the group which means the length of the time slot cannot be shortened. The conflicting requirements of shorter time slot and higher computation results in systems that not practical for large groups.

Our Contribution

We consider the same problem as Fiat and Tassa: detecting *all* traitors when traitors re-broadcast the content. However we propose a different solution which does not use the feedback signal for mark allocation and so (i) will not be vulnerable to delayed rebroadcast attack, and (ii) does not require real-time computation for mark allocation and so allow very short time slots.

Similar to Fiat and Tassa, we mark consecutive segments of the content and detect the feedback from the channel. However unlike their scheme the allocation of marks to users in each segment is pre-determined and follows a fixed table. We call this system a *sequential traitor tracing* scheme to emphasise the fact that more than one step is required, and at the same time differentiate it from dynamic schemes. In a dynamic scheme allocation of the marks and delivery of marked versions to users is *after* receiving the feedback signal. That is only after observing the re-broadcasted segment from previous round, re-partitioning of the user set, and calculation and delivery the required keys followed by the marked segment for the next segment can be performed. In our approach because the mark allocation for each segment does not require the feedback of the previous segment, the system will be much more efficient. This means that because no real-time computation is required the length of a segment can be chosen very short. The cost paid for the added security and real-time efficiency could be higher bandwidth or convergence rate (number of steps). However because of shorter time slots the total convergence time could remain comparable or even reduced.

We give a formal definition of the new scheme and derive a bound that relates the number of versions (communication efficiency) and convergence time of the algorithm. We give two general constructions that can be used with any robust watermarking system. In particular we give a construction that allows tracing of all up to p traitors in a group of $f(q)$ users in at most $(p(p+1))^t$ steps and requires only q versions, where $f(q)$ is a polynomial of q with degree 2^t, and a second construction using error correcting codes that allows all up to p traitors in a group of N users be traced. This construction requires $2p$ versions and has the convergence time equal to $8p \log N + p$ steps which is the same as the best dynamic scheme. We will show that both of these constructions are general and can be used with p-frameproof codes and p-traceability schemes to construct systems for large groups.

The paper is organised as follows. In section 2 we give the required definitions and review the known results. In section 3 we introduce our model and derive bounds relating efficiency parameters of the system. The two constructions follow in section 4. In section 5 we evaluate our results and their extensions.

2 Preliminaries

In this section we briefly review relevant definitions and results.

Broadcast Encryption

In a *broadcast encryption system* a centre generates a set of *base keys*, and assigns a subset of these keys to every user as his *personal key* such that at a later time it can broadcast an encrypted message that is only accessible to a privileged subgroup and users who are not in the subgroup cannot decrypt the message. The privileged subgroup is not fixed and may be one of a set of possible authorised subsets. *Resilience* of a broadcast encryption system is measured by a parameter k which is the the size of largest colluding group, disjoint from the privileged set, who cannot learn the message.

Marking Digital Content

Marking a digital object has been initially studied in the context of frame-proof codes [2]. Consider a set of n users $\{1, 2, \cdots, n\}$. Let Σ be an alphabet. An (ℓ, n)-code is a set $\Gamma = \{c_1, c_2, \cdots, c_n\} \subseteq \Sigma^\ell$. Let T be a coalition of users, and assume $i \in \{1, 2, \cdots, \ell\}$ is a position. We say that position i is *undetectable* for T if the words assigned to T match in their i^{th} position. Denote by R the set of undetectable positions for T . Define the *feasible set* of T as consisting of all $w \in \Sigma^\ell$ such that w and the words assigned to T are matched in R. Denote by $F(T)$ the feasible set of T.

A code Γ is called *p-frameproof* if every set $W \subset \Gamma$, of size at most p, satisfies $F(W) \cap \Gamma = W$. Frameproof codes are useful if a software, or a binary file needs to be marked. Detection of a mark requires the mark embedded in the content to be their exact stored values. An extended definition of frameproof code [3], [9] allows the marks to be also deleted however this is again for a mark in the codeword.

Protection of video and audio signal content is through *watermarking systems* that cannot be strictly modelled by a frameproof code. We define a *watermarking code* for audio and video content as a collection of distinct codewords (or marks), $C = \{c_1, c_2, \cdots, c_r\}$, and two algorithms I for *watermark insertion*, and D for *watermark detection* [11]. The insertion algorithm takes a codeword c_i and a content m and produces a *marked version* m_i. The detection algorithm takes a content m' and a codeword c_i and produces a *true/false* value depending on success or failure of the detection. In watermark codes a codeword c_i is a distinct *whole* mark and collusion of users may either convert the whole mark into a different mark, or completely remove it. In practice the former type of attack has negligible success probability and it is enough to consider the latter one. A watermarking code is *robust* if no combination of marked objects $\mu = \{m_{i_1}, \cdots, m_{i_\ell}\}$ can produce another marked object $m_i \notin \mu$, or delete the mark. Robust watermarking codes model robust watermarking schemes such as Cox et al [5] and their properties match properties of watermarking systems in practice. They are more general than frameproof codes because there is no restriction on an attacker's operation (feasible set in frameproof codes). This is a realistic model as in practice watermarks are inserted in many different ways and can be subjected to a wide range of attacks.

Traitor Tracing

Similar to a broadcast encryption system the content provider generates a base set of keys, and assigns subsets of it to each user. The subset of keys received by a user i forms his *personal key* or his *decoder key*, and is denoted by U_i. By holding U_i user i will be able to view the content. A colluding set of users can construct a *pirate decoder* which contains a subset of their keys. When a pirate decoder F is found, $|F \cap U_i|$ for all i is calculated and if $|F \cap U_i| \geq |F \cap U_j|$ for all j, then user i is called an *exposed user*. Following Definition 1.2 in [9], a scheme is called a *p-traceability scheme* if whenever a pirate decoder F is produced by T and $|T| \leq p$, the exposed user is a member of the coalition T.

Suppose there are ℓ base keys $\{k_1, k_2, \cdots, k_\ell\}$ and n users. Then a p-traceability scheme can be defined by an 0-1 matrix of size $n \times \ell$, such that its $(i,j)^{th}$ element is 1 if and only if user i has key k_j in his decoder.

In a *static traceability scheme* keys are allocated once and remain unchanged through the operation of the system.

FT Scheme

In FT scheme *content* consists of a number of *segments*, for example one minute of a video or a movie. For each segment a number of *variants* using a *robust watermarking code*, such as *spread spectrum technique* of Cox et al [5], is constructed. A fundamental assumption of the system is that because of the robustness of the watermarking system the re-broadcasted version is one of the versions owned by the members of the colluder group.

For each segment the user set is partitioned into r subsets, each subset receiving the same version. Each user has some key information that allows the content provider to securely give him a session key for his version, or use a broadcast encryption system to securely deliver his version. They proved that for tracing p traitors in any traceability system at least $p+1$ versions must be used, and gave algorithms that used $p+1$ and $2p+1$ versions and required $O(3^p p \log n)$ and $O(p \log n)$ steps to converge, respectively. Their algorithms were improved by Berkman et al [1] who showed an algorithm with $O(p \log n + p^2/c)$ step for convergence and using $p+c+1$ versions, and a second one with $O(p \log n)$ and $pc+1$ versions. Again the main emphasis of their work was to find schemes that allow best convergence when close to minimum possible number of versions is used.

Sequential Traitor Tracing

One of the main drawbacks of FT model is the high real-time computation required for allocation of marks for each segment. Because this computation depends on the feedback from the previous segment it must be performed in real-time and there is no possibility for precomputation which implies that the length of a step must be chosen long enough for the required computation. Because the overall *convergence time* is a product of the number of steps and the *length of the step*, and because more complex algorithms with small number of steps require more computation and so longer segment, it is important to optimise the number of steps versus the required real-time computation.

We propose a sequential tracing scheme in which mark allocation is predetermined and so real-time processing in each segment is not required. Although the mark allocation changes in each step but because it is according to a known table, all the required computation for most of the allocation can be performed as pre-computation. The feedback signal is *only* used for detection of traitors. Although the system has a more limited use of the feedback compared to the full dynamic model of FT and so can be expected to require more steps for convergence, but because of much smaller computation in real-time a much shorter length for segments is possible and so the overall convergence can be expected to be lower.

Before presenting our model, we note that the following attack is outside both FT and our model.

- *Framing a user by re-broadcasting his version:* If a broadcast encryption is used for secure delivery of segments, then a colluder subgroup may construct a pirate decoder and obtain the version v_p of an innocent user. If such a version is re-broadcasted all detection algorithms fail. This would be a feasible attack if the broadcast encryption system used for sending a version to the target subgroup does not provide traceability. Same effect can be obtained if colluders can break the underlying watermarking system and construct the version of another user. We noted that this in general is a very unlikely event.

This means that if a broadcast encryption is used it must be able to trace traitors.

3 The Model

Let $\mathcal{U} = \{1, 2, \cdots, N\}$ denote the set of users. A user i has some secret key information, U_i, that allows the content provider to identify him and send him a particular version. U_i could be a set of keys in a broadcast encryption scheme or a secret used to encrypt the session key of the user. There is a *mark allocation table* M with N rows and d columns where $M(i, j)$ is the mark allocated to user i in segment j. In each segment the content provider sends the j^{th} segment to users according to column j of M and observes the feedback. Traitors can be detected by examining the sequence of feedback signals and after d feedbacks it is possible to trace all the traitors: that is the tracing algorithm *converges.* When a traitor is found, he is *disconnected.* This is by excluding the user from the broadcast encryption system in all future segments. That is, if i is detected as a traitor in segment j, then from segment $j + 1$, his reception of segment $M(i, k), k \geq j + 1$, will be blocked.

Assume there is a probability distribution on $U_1 \times U_2 \times \cdots \times U_N$. For $X = \{i_1, i_2, \cdots, i_j\} \in 2^{\mathcal{U}}$ denote by $U_X = U_{i_1} \times U_{i_2} \times \cdots \times U_{i_j}$, $i_1 < i_2 < \cdots < i_j$, the set of secret information given to X.

There is a watermarking code, $C = \{c_1, \cdots, c_r\}$, used to mark *segments* of the *protected content.* A segment with a valid mark is called a *variant.* There is a *feedback sequence* $F = ()$ which is initialised to the empty sequence. Let $T \subset \mathcal{U}$ be a set of *traitors* and $\mathcal{P}^{(j)}$ denote the set of all privileged users in the j^{th} segment. The set $\mathcal{P}^{(j)}$ is partitioned into $\mathcal{P}^{(j)} = P_1^{(j)} \cup P_2^{(j)} \cup \cdots \cup P_r^{(j)}$, and each subset $P_i^{(j)}$ is allocated a version marked by c_i. Let $\mathcal{V}^{(j)}$ denote the set of all possible versions in segment j. In a segment j the content provider uses a vector of r versions, $\mathcal{V}^{(j)} = (V_1^{(j)}, V_2^{(j)}, \cdots, V_r^{(j)}) \in \mathcal{V}^{(j)r}$ where $V_i^{(j)} \neq V_l^{(j)}$, $i, l \in \{1, \cdots, r\}, i \neq l$. There is a probability distribution on $\mathcal{V}^{(j)}$. In each segment j there is a feedback signal $f_j = c_i$ for some i such that $P_i^{(j)} \cap T \neq \emptyset$, which is appended to F_{j-1} to construct $F_j = (f_1, \cdots, f_j)$. This sequence is used to trace traitors. A feedback sequence F_d is *p-consistent* if it can be generated by a colluder set of size at most p.

Definition 1. *A sequential (p, d)-traceability scheme is a family of partitions $\mathcal{P}^{(j)} = P_1^{(j)} \cup P_2^{(j)} \cup \cdots \cup P_r^{(j)}$, $j = 1, 2, \cdots, d$, with the following properties*

1. *In each segment j, each user receives a version. Formally, for each $U \in P_i^{(j)}$, $H(\mathcal{V}_i^{(j)} \,|\, U) = 0$.*
2. *In each segment j, a group of users which is disjoint from $P_i^{(j)}$ cannot have any information on versions of members of $P_i^{(j)}$ even if they use all previous feedbacks. Formally, for each $X \in 2^{\mathcal{U}}$, with $X \cap P_i^{(j)} = \emptyset$, $H(\mathcal{V}_i^{(j)} \,|\, U_X, F_{j-1})$ $= H(\mathcal{V}_i^{(j)})$.*
3. *After d rounds all up to p traitors can be detected. Formally, any p-consistent feedback sequence F_d determines a unique colluder set of at most p, that is $H(\mathcal{U}_T \,|\, F_d) = 0$.*

The following proposition shows that in a segment j, a user does not have any information about the version assigned to another user belonging to a group different from his.

Proposition 1. *Let j, b be integers, $b \leq r$, $X_i \subseteq P_i^{(j)}$, $1 \leq i \leq b$ such that $X_i \cap X_k = \emptyset$ for every pair $i \neq k, 1 \leq i, k \leq b$. Then*

$$H(\mathcal{V}_i^{(j)} \,|\, \mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)}) = H(\mathcal{V}_i^{(j)}).$$

The following theorem gives a lower bound on the number of rounds required to detect all traitors.

Theorem 1. *Suppose in a (p, d)-traceability scheme there are N users, at most p traitors and r versions. Then*

$$d \geq p \log_r N.$$

4 Constructions

In this section we give two constructions for sequential traceability schemes from watermarking codes, one using a special class of functions and the second one using error correcting codes.

4.1 A Construction Using a Function Family

This construction uses a robust watermarking code with n codewords and results in a sequential scheme that identifies at least one of the traitors (at most p traitors) in $p^2 + 1$ steps, and *all* the traitors in at most $p^2 + p$ steps. The scheme converges in $p(p + 1)$ steps and so convergence time is independent of the size of the group. This is at the expense of higher number of versions and so less communication efficiency. We will show (section 5) that the scheme can be repeatedly used to increase the number of users with the same number of versions (communication efficiency) while increasing the number of rounds.

Suppose we have a robust watermarking code with n codewords $c_1, c_2, \cdots, c_n$, and let M denote the $n \times 1$ matrix with c_i as its i^{th} row.

Consider a collection of mappings, $\Phi = \{\phi_{ij} : 1 \leq i \leq b, 1 \leq j \leq m\}$,

$$\phi_{ij} : \{1, 2, \cdots, n\} \to \{1, 2, \cdots, n\},$$

that satisfy the following two properties:

(P1) For each j and each pair of the first index (i_1, i_2) with $i_1 \neq i_2$, we have $\phi_{i_1 j}(x) \neq \phi_{i_2 j}(x)$ for all $x \in \{1, 2, \cdots, n\}$.

(P2) For each pair of the first index (i_1, i_2) and each pair of the second index (j_1, j_2) with $j_1 \neq j_2$, we have $\phi_{i_1 j_2}(x) \neq \phi_{i_2 j_2}(y)$ provided that $\phi_{i_1 j_1}(x) = \phi_{i_2 j_1}(y)$.

Given Φ and a watermarking code with n codewords define a matrix $\widetilde{M}$ as follows.

$$\widetilde{M} = \begin{pmatrix} M & M_1 & \phi_{11}(M) & \phi_{12}(M) & \cdots & \phi_{1m}(M) \\ M & M_2 & \phi_{21}(M) & \phi_{22}(M) & \cdots & \phi_{2m}(M) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ M & M_b & \phi_{b1}(M) & \phi_{b2}(M) & \cdots & \phi_{bm}(M) \end{pmatrix}, \tag{1}$$

where

$$M_i = \begin{pmatrix} c_i \\ c_i \\ \vdots \\ c_i \end{pmatrix} \text{ and } \phi_{ij}(M) = \begin{pmatrix} c_{\phi_{ij}(1)} \\ c_{\phi_{ij}(2)} \\ \vdots \\ c_{\phi_{ij}(n)} \end{pmatrix}$$

are $n \times 1$ matrices.

$\widetilde{M}$ has b *block rows*, each block row contains n rows, each row assigned to a user in the sequential scheme. Denote by (r, k) the user holding the k^{th} row of the r^{th} block row.

$\widetilde{M}$ has $m+2$ *block columns* where the j^{th} block column contains the marks that will be allocated to users in segment j.

The scheme works as follows. In each segment (except for the second one, if $b < n$) there are n versions that are marked with codewords of the watermarking code. For each segment the set of users $\mathcal{U}$ is divided into b equal size subgroups and members of each subgroup (of size n) will receive a version marked by the same codeword of the watermarking code. In segment j, the centre observes the feedback and detects a codeword $f_j \in \{c_1, \cdots, c_n\}$. It updates the feedback sequence and obtains F_j. F_j is compared with the rows of $\widetilde{M}$. If a row with $p+1$ 'match' is found, a traitor is detected.

Suppose the center has a feedback sequence $F_d = (f_1, f_2, \cdots, f_d)$, where f_j corresponds to the j^{th} segment. We say F_d *matches* a row $R = (X_1, X_2, \cdots, X_{m+2})$ of $\widetilde{M}$ in t positions, $t < d$, if there exist indices $j_1 < j_2 < \cdots < j_t$ such that $f_{j_1} = X_{j_1}, \cdots, f_{j_t} = X_{j_t}$.

The algorithm can be described as follows. Let ϕ_{i0} denote a constant mapping from $\{1, 2, \cdots, n\}$ to $\{1, 2, \cdots, n\}$, that is $\phi_{i0}(x) = i$ for all $x \in \{1, 2, \cdots, n\}$, and $i = 1, 2, \cdots, b$. In the beginning suppose each user has a version according to the first block column in (1). Then the algorithm is as follows.

The Algorithm

```
1. Set $h = 0$, and $F$ to an empty list, $F = ()$.
2. Repeat while $h \leq m$:
    (a) For $r = 1, \cdots, b, k = 1, \cdots, n$, send a variant of segment $h$ marked with
        $c_{\phi_{rh}(k)}$ to the user corresponding to the row $(r-1)n + k$.
    (b) Receive the feedback and extract the mark $f_h$.
    (c) Append the feedback $f_h$ to the list $F$.
    (d) compare $F$ with the first $h+1$ block columns of $\widetilde{M}$. If a row matches $F$
        in $p+1$ block columns disconnect the corresponding user.
    (e) Increment $h$.
```

Theorem 2. *The scheme described above can correctly detect all traitors.*

The proof of the theorem is based on the following Lemma.

Lemma 1. *Suppose there are at most p traitors. Let*

$$F = (f_{j_1}, f_{j_2}, \cdots, f_{j_d}), \quad d \geq p+1, \tag{2}$$

be a p-consistent feedback sequence. If (2) matches the $(r-1)n + k^{th}$ row (row (r, k) for short) of (1) in $p+1$ positions, then no collusion excluding (r, k), of at most p users, can produce (2).

Proof. Otherwise suppose collusion T produces (2), $|T| \leq p$, $(r,k) \notin T$. By assumption there are $p+1$ positions where (r,k) matches (2). Among these $p+1$ positions there exist two of them, say j_{t_1}, j_{t_2}, such that some $(r',k') \in T$ matches (2), and hence matches (r,k), at j_{t_1}, j_{t_2}. By (1) we know that

- either $c_k = c_{k'} = f_{j_{t_1}}$, $c_{\phi_{r,j_{t_2}}(k)} = c_{\phi_{r',j_{t_2}}(k')} = f_{j_{t_2}}$, or
- $c_{\phi_{r,j_{t_1}}(k)} = c_{\phi_{r',j_{t_1}}(k')} = f_{j_{t_1}}$, $c_{\phi_{r,j_{t_2}}(k)} = c_{\phi_{r',j_{t_2}}(k')} = f_{j_{t_2}}$.

The first case implies that $k = k'$ and $\phi_{r,j_{t_2}}(k) = \phi_{r',j_{t_2}}(k')$, and hence $\phi_{r,j_{t_2}}(k) = \phi_{r',j_{t_2}}(k)$, which contradicts (P1) since we know $r \neq r'$ because $k = k'$, and $(r,k) \neq (r',k')$. The second case implies that $\phi_{r,j_{t_1}}(k) = \phi_{r',j_{t_1}}(k')$ and $\phi_{r,j_{t_2}}(k) = \phi_{r',j_{t_2}}(k')$, which contradicts (P2) as $t_1 \neq t_2$.

Proof of Theorem 2: (*sketch*) From lemma 1 we know that when F matches a row of $\widetilde{M}$ in $p+1$ positions then a traitor can be identified. The traitor is *disconnected* so that he cannot decrypt future segments and the system continues as before. This means that p traitors can be captured in at most $p(p+1)$ steps.

Existence of Φ

The construction in section 4.1 relies on the existence of a function family Φ that satisfies property (P1) and (P2). The number of users in the resulting sequential scheme is bn and so is proportional to the size of the function family. In the following we give a construction for Φ satisfying properties (P1) and (P2).

Theorem 3. *Let q be a prime number. There exists a function family $\Phi = \{\phi_{ij} : 1 \leq i,j \leq (q-1)/2\}$ that satisfies properties (P1) and (P2).*

Proof. Let $\mathbf{F}_q$ be a field of q elements, $\mathbf{F}_q^*$ be the set of non-zero elements of $\mathbf{F}_q$. For $i,j \in \{1,2,\cdots,(q-1)/2\}$ define

$$\phi_{ij} : \mathbf{F}_q^* \to \mathbf{F}_q^*$$

such that $\phi_{ij}(x) = (i+j)x$. Obviously $\phi_{i_1 j}(x) \neq \phi_{i_2 j}(x)$ for all $x \in \mathbf{F}_q^*$ provided $i_1 \neq i_2$ and so (P1) is satisfied. Now assume $\phi_{i_1,j_1}(x) = \phi_{i_2,j_1}(y)$. Then using the definition of ϕ_{ij}, we have $i_1 x - i_2 y = j_1(y-x)$. So for every $j_2 \neq j_1$ we have $i_1 x - i_2 y \neq j_2(y-x)$, which implies that $\phi_{i_1,j_2}(x) \neq \phi_{i_2,j_2}(y)$. Hence (P2) is satisfied.

An example of this construction is given in Appendix 2.

Discussion

Combining the above Φ and a watermarking code with $q-1$ codewords we obtain a sequential scheme in which (i) $q-1$ variants are used, (ii) $(q-1)^2/2$ users are accommodated, and (iii) at most p^2+p rounds are needed for detection of all p traitors.

In choosing q, the number of variants, we must consider N, the total number of users in the final system and p, the maximum number of traitors. We must have $(q-1)^2/2 \geq N$ and also $p^2+p \leq 2+(q-1)/2$ and so $q \geq \max(1+\sqrt{2N}, 2p^2+2p-3)$.

4.2 A Construction Using Error-Correcting Codes

We can construct a sequential scheme by combining watermarking codes and error-correcting codes. The method is similar to the one given in [3] for constructing frameproof codes.

Let $\mathcal{C}_1$ be an $(L, N, D)_n$-ECC error-correcting code over an alphabet of size n with N codewords each of length L, and minimum distance D. Let $\mathcal{C}_2 = \{c_1, c_2, \cdots, c_n\}$ be a watermarking code. Define the *composition* of $\mathcal{C}_1$ and $\mathcal{C}_2$, denoted by $\Gamma(\mathcal{C}_1, \mathcal{C}_2)$, as a collection of strings over $\mathcal{C}_2$ obtained as

$$C_v = c_{a_1} \parallel c_{a_2} \parallel \cdots \parallel c_{a_L}, \tag{3}$$

for all codewords v, $v = a_1 a_2 \cdots a_L \in \mathcal{C}_1$. Here $\parallel$ means concatenation of strings.

Theorem 4. *Suppose we have $\mathcal{C}_1 = \{c_1, c_2, \cdots, c_n\}$ a watermarking code, and $\mathcal{C}_2$, a $(L, N, D)_n$-ECC. If*

$$D > (1 - \frac{1}{p})L,$$

then $\Gamma(\mathcal{C}_1, \mathcal{C}_2)$ defines a sequential (p, d)-traceability scheme in which all, up to p, traitors can be traced in at most $p(L - D + 1)$ steps.

Proof. (*sketch*) Let $\mathcal{C}_1 = \{c_1, c_2, \cdots, c_n\}$, then

$$\Gamma(\mathcal{C}_1, \mathcal{C}_2) = \{(c_{a_1}, c_{a_2}, \cdots, c_{a_L}) \,|\, (a_1, a_2, \cdots, a_L) \in \mathcal{C}_2\}.$$

Assign each string in $\Gamma(\mathcal{C}_1, \mathcal{C}_2)$ to a user: that is let the marks in the string be the user's L successive marks. Suppose at most p traitors contribute marks to the feedback sequence,

$$F_d = (f_1, f_2, \cdots, f_d). \tag{4}$$

When $d \geq p(L - D + 1)$, there is a traitor whose mark sequence coincides with (4) in at least $L - D + 1$ places. This user can be detected at this stage. To disconnect all up to p traitors, at most $p(L - D + 1)$ steps are required. Here $D \geq (1 - \frac{1}{p})L + 1$ guarantees that $L \geq p(L - D + 1)$.

The following theorem shows that error correcting codes with suitable parameters, as required in Theorem 4, exist.

Theorem 5. *(Lemma III.3 of [3]) For any positive integers p, N, let $L = 8p \log N$. Then there exists a $(L, N, D)_{2p}$-ECC where $D > (1 - \frac{1}{p})L$.*

Now suppose we have a watermarking code $\mathcal{C}_1$ with $2p$ codewords. Let $\mathcal{C}_2$ be an $(L, N, D)_{2p}$-ECC and $D > (1 - \frac{1}{p})L$. From Theorem 4, the composition of the two codes is a (p, d)-traceability scheme. In this scheme $2p$ versions are used, and the number of rounds to detect all traitors is no more than $p(L - D + 1)$. So we have

$$\begin{aligned} d \leq p(L - D + 1) &< pL - p(1 - \frac{1}{p})L + p \\ &= L + p = 8p \log N + p \ \ \text{(from Theorem 5)} \end{aligned}$$

This is the same order as $O(p \log N + p^2/c)$ which is the best known result [1] for $p + c + 1$ versions.

5 Comparison and Discussion

It is not difficult to show that (proof is omitted) the mark allocation of the schemes in section 4 defines a frameproof code.

A very interesting aspect of the constructions given in section 4 is that if watermarking code in these constructions is replaced by a frameproof code or a static traceability scheme, the resulting code will be a frameproof code or traceability scheme, respectively.

The following theorems summarises these results.

Theorem 6. *Let Φ be a family of functions satisfying properties (P1) and (P2), $m > p - 2$.*

1. *If $\mathcal{C} = \{c_1, c_2, \cdots, c_n\}$ is a p-frameproof code, then the code with incidence matrix given by $\widetilde{M}$ as defined in (1) is a p-frameproof code.*
2. *If $\mathcal{C} = \{c_1, c_2, \cdots, c_n\}$ is a p-traceability scheme, then the code with incidence matrix given by $\widetilde{M}$ as defined in (1) is a p-traceability scheme.*

Repeated use of the above theorem results in a p-frameproof code (p-traceability scheme) with the following parameters: (i) length of the codeword is $(p+1)^t \ell$, and (ii) the number of the codewords is $f(n)$, a polynomial of degree 2^t, assuming that the original code has n codewords of length ℓ.

Another interesting observation is that starting from a watermarking code and through the application of constructions in section 4 we will obtain a frameproof code and then using the above theorem we can construct a frameproof code for a much larger group. However because of the underlying watermarking code the resulting matrix can be used for a sequential traceability system. This proves the following corollary.

Corollary 1. *There is a sequential (p, d)-traceability scheme for $f(q)$ users using q marks and with $d = (p(p+1))^t$, where $f(q)$ is a polynomial in q with degree 2^t.*

The results will hold even if we relax the restriction on the watermarking code and allow collusion of users to remove the watermark. As noted earlier this is the main type of attack in watermarking systems.

Theorem 7. *Let $\mathcal{C}$ be an $(L, N, D)_n$-ECC with $D > (1 - \frac{1}{p})L$.*

1. *If $\mathcal{C}_1 = \{c_1, c_2, \cdots, c_n\}$ is a p-frameproof code, then the composition of $\mathcal{C}$ and $\mathcal{C}_1$ is a p-frameproof code.*
2. *If $\mathcal{C}_2 = \{c_1, c_2, \cdots, c_n\}$ is a p-traceability scheme, then the composition of $\mathcal{C}$ and $\mathcal{C}_2$ is a p-traceability scheme.*

We note that 1 in Theorem 7 was first proved in Lemma III.2 in [3] and is included here for the sake completeness.

5.1 Computational Efficiency

The most attractive feature of the above model is the reduced real-time computation. Because allocation of marks is static most of the required keys can be pre-computed and distributed to the users in one initial block. Key updates are required when a traitor is found and needs to be disconnected. An important result of this efficiency is that it is possible to reduce the length of a segment and hence reduce the overall convergence time.

The detection algorithm is also very efficient. This is because we only require detection of partial match. For this it is only required to keep a counter for each user (a row of the mark allocation matrix) that counts the number of matches between the feedback sequence and that row. When a feedback f_j is received the counter for all the rows that have f_j in their j^{th} position are incremented.

5.2 Time/Bandwidth Trade-Off

Two important parameters of dynamic traitor tracing schemes are (i) the number of marks, r, which determines the communication efficiency of the system, and (ii) the number of steps, d, required for convergence, that is finding all the traitors. Fiat and Tassa, and later Berkman et al [1] concentrated on the communication efficiency and so finding the efficient algorithms when r is close to its to theoretical minimum $p+1$. Berkman et al showed that if $r = pc + 1$ versions are used it is possible to find the traitors in $O(p \log_c N)$ rounds, and if $p + c + 1$ versions are used it is possible to find the traitors in $O(p^2/c + p \log N)$ rounds. Our first scheme guarantees convergence in $(p(p+1))^t$ steps while the number of users can be about $p^{2^{t+1}}$. This scheme uses $q(\geq p^2 + p - 2)$ versions.

For the second scheme the convergence time is at most $8p \log N + p$ which is of the same order as the best result of [1]. It is important to note that by reducing the segment length the overall convergence time of our scheme would be expected to be lower.

5.3 Conclusions

Sequential traceability schemes can be seen as a step between static and dynamic schemes. The attack model in sequential schemes and dynamic schemes are the same and is different from a static traceability scheme. Also the goal of the former two types of systems are the same (tracing all traitors) and is different from static schemes. Sequential schemes do not have the flexibility of dynamic schemes and so in general could require higher bandwidth and/or higher number of convergence steps. However they provide security against delayed rebroadcast attack and are also practically attractive because they do not require real-time computation. We showed a construction that is as good as the best known dynamic construction and so in terms of efficiency measures competes well with dynamic schemes. We also showed that sequential traceability schemes are closely related to frameproof codes and so constructions from frameproof codes can be used for sequential schemes.

Although we gave a bound on the number of steps for convergence, but deriving tight bounds and developing schemes that achieve the bound need further research.

References

1. O. Berkman, M. Parnas, and J. Sgall. Efficient dynamic traitor tracing. to appear at SODA 2000, 2000.
2. D. Boneh and J. Shaw. Collusion-secure fingerprinting for digital data. In *Advances in Cryptology - CRYPTO'95, Lecture Notes in Computer Science*, volume 963, pages 453–465. Springer-Verlag, Berlin, Heidelberg, New York, 1995.
3. D. Boneh and J. Shaw. Collusion-secure fingerprinting for digital data. *IEEE Transactions on Information Theory*, Vol.44 No.5:1897–1905, 1998.
4. B. Chor, A. Fiat, and M. Naor. Tracing traitors. In *Advances in Cryptology - CRYPTO'94, Lecture Notes in Computer Science*, volume 839, pages 257–270. Springer-Verlag, Berlin, Heidelberg, New York, 1994.
5. I. Cox, J. Killian, T. Leighton, and T. Shamoon. Secure spread spectrum watermarking for multimedia. *IEEE Transaction on Image Processing*, Vol. 6 no. 12:1673–1687, 1997.
6. A. Fiat and M. Naor. Broadcast encryption. In *Advances in Cryptology-CRYPTO'93, Lecture Notes in Computer Science*, volume 773, pages 480–491. Springer-Verlag, Berlin, Heidelberg, New York, 1994.
7. A. Fiat and T. Tassa. Dynamic traitor tracing. In *Advances in Cryptology-CRYPTO'99, Lecture Notes in Computer Science*, volume 1666, pages 354–371. Springer-Verlag, Berlin, Heidelberg, New York, 1999.
8. E. Gafni, J. Staddon, and Y.L. Yin. Efficient methods for intergrating traceability and broadcast encryption. In *Advances in Cryptology-CRYPTO'99, Lecture Notes in Computer Science*, volume 1666, pages 372–387. Springer-Verlag, Berlin, Heidelberg, New York, 1999.
9. D. Stinson and R. Wei. Combinatorial properties and constructions of traceability schemes and framproof codes. *SIAM Journal on Discrete Mathematics*, 11:41–53, 1998.
10. D. R. Stinson and R.Wei. Key preassigned traceability schemes for broadcast encryption. In *Proceedings of SAC'98, Lecture Notes in Computer Science*, volume 1556, pages 144–156. Springer-Verlag, Berlin, Heidelberg, New York, 1999.
11. M. Swanson, M Kobayashi, and A. Tewfik. Multimedia data-embedding and watermarking technologies. *Proceedings of IEEE*, Vol.86 no.6:1064–1087, 1998.

Appendix 1

Proof of Proposition 1 Take a subset Y with $Y \cap X_i = \emptyset$ and $Y \cap X_k \neq \emptyset$ for $k \neq i$. Suppose F_{j-1} is a feedback sequence. Then we have

$$\begin{aligned} & I(\mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)}; \mathcal{V}_i^{(j)} \,|\, U_Y, F_{j-1}) \\ &= H(\mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)} \,|\, U_Y, F_{j-1}) \\ &\quad - H(\mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)} \,|\, \mathcal{V}_i^{(j)}, U_Y, F_{j-1}) \end{aligned}$$

$$\begin{aligned}
&\le H(\mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)} \,|\, U_Y, F_{j-1}) \\
&\le \sum_{k \neq i} H(\mathcal{V}_k^{(j)} \,|\, U_Y, F_{j-1}) \\
&= 0 \ \text{ (because of } Y \cap X_k \neq \emptyset)
\end{aligned}$$

So $I(\mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)}; \mathcal{V}_i^{(j)} \,|\, U_Y, F_{j-1}) = 0$. Note that

$$\begin{aligned}
0 &= I(\mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)}; \mathcal{V}_i^{(j)} \,|\, U_Y, F_{j-1}) \\
&= I(\mathcal{V}_i^{(j)}; \mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)} \,|\, U_Y, F_{j-1}) \\
&= H(\mathcal{V}_i^{(j)} \,|\, U_Y, F_{j-1}) - H(\mathcal{V}_i^{(j)} \,|\, \mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)}, U_Y, F_{j-1}).
\end{aligned}$$

Then we get

$$\begin{aligned}
H(\mathcal{V}_i^{(j)}) &\ge H(\mathcal{V}_i^{(j)} \,|\, \mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)}) \\
&\ge H(\mathcal{V}_i^{(j)} \,|\, \mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)}, U_Y, F_{j-1}) \\
&= H(\mathcal{V}_i^{(j)} \,|\, U_Y, F_{j-1}) \\
&= H(\mathcal{V}_i^{(j)}) \ \text{ (because of } Y \cap X_i = \emptyset).
\end{aligned}$$

So $H(\mathcal{V}_i^{(j)} \,|\, \mathcal{V}_1^{(j)}, \cdots, \mathcal{V}_{i-1}^{(j)}, \mathcal{V}_{i+1}^{(j)}, \cdots, \mathcal{V}_b^{(j)}) = H(\mathcal{V}_i^{(j)})$. The proposition is proved.

Proof of Theorem 1 Suppose T is a set of p traitors. Let $F_d = (f_1, f_2, \cdots, f_d)$ be the corresponding feedback sequence. Then we have

$$\begin{aligned}
H(\mathcal{U}_T) &= H(\mathcal{U}_T, F_d) - H(F_d \,|\, \mathcal{U}_T) \\
&= H(F_d) + H(\mathcal{U}_T \,|\, F_d) - H(F_d \,|\, \mathcal{U}_T) \\
&= H(F_d) - H(F_d \,|\, \mathcal{U}_T), \ \text{ (by 1 of definition 1)} \\
&\le H(F_d) = H(f_1, f_2, \cdots, f_d) \\
&\le H(f_1) + H(f_2) + \cdots + H(f_d) \\
&\le d \log r
\end{aligned}$$

Note that $H(\mathcal{U}_T) = pH(\mathcal{U})$. So $pH(\mathcal{U}) \le d \log r$. It implies that $p \log N \le d \log r$, here N, d, r is the number of total users, the number of rounds to detect traitors and the number of versions, respectively. The theorem is proved.

Proof of Theorem 6 Let $T = \{t_1, t_2, \cdots, t_c\}, c \le p$, be a set of traitors. Suppose they collude to frame a user $U_i \notin T$. Since $m > p - 2$, using Pigeonhole Principle U_i must match one $t_j \in T$ in more than one place. This contradicts (P2). So T can not frame other users. The second result can be proved in a similar way.

Appendix 2

An Example

Suppose we want to provide protection for up to 50 users against collusion of up to 2 colluders ($p = 2$). We need $q = 11$ in theorem 3.

For simplicity we use i instead of $\underset{\sim}{c_i}$. That is we only list the indices of codewords in watermarking code. Then $\widetilde{M}$ consists of the following blocks.

$(1,1):$	1	1	2	3	4	5	$(2,1):$	1	2	3	4	5	6	$(3,1):$	1	3	4	5	6	7
$(1,2):$	2	1	4	6	8	10	$(2,2):$	2	2	6	8	10	1	$(3,2):$	2	3	8	10	1	3
$(1,3):$	3	1	6	9	1	4	$(2,3):$	3	2	9	1	4	7	$(3,3):$	3	3	1	4	7	10
$(1,4):$	4	1	8	1	5	9	$(2,4):$	4	2	1	5	9	2	$(3,4):$	4	3	5	9	2	6
$(1,5):$	5	1	10	4	9	3	$(2,5):$	5	2	4	9	3	8	$(3,5):$	5	3	9	3	8	2
$(1,6):$	6	1	1	7	2	8	$(2,6):$	6	2	7	2	8	3	$(3,6):$	6	3	2	8	3	9
$(1,7):$	7	1	3	10	6	2	$(2,7):$	7	2	10	6	2	9	$(3,7):$	7	3	6	2	9	5
$(1,8):$	8	1	5	2	10	7	$(2,8):$	8	2	2	10	7	4	$(3,8):$	8	3	10	7	4	1
$(1,9):$	9	1	7	5	3	1	$(2,9):$	9	2	5	3	1	10	$(3,9):$	9	3	3	1	10	8
$(1,10):$	10	1	9	8	7	6	$(2,10):$	10	2	8	7	6	5	$(3,10):$	10	3	7	6	5	4

$(4,1):$	1	4	5	6	7	8	$(5,1):$	1	5	6	7	8	9
$(4,2):$	2	4	10	1	3	5	$(5,2):$	2	5	1	3	5	7
$(4,3):$	3	4	4	7	10	2	$(5,3):$	3	5	7	10	2	5
$(4,4):$	4	4	9	2	6	10	$(5,4):$	4	5	2	6	10	3
$(4,5):$	5	4	3	8	2	7	$(5,5):$	5	5	8	2	7	1
$(4,6):$	6	4	8	3	9	4	$(5,6):$	6	5	3	9	4	10
$(4,7):$	7	4	2	9	5	1	$(5,7):$	7	5	9	5	1	8
$(4,8):$	8	4	7	4	1	9	$(5,8):$	8	5	4	1	9	6
$(4,9):$	9	4	1	10	8	6	$(5,9):$	9	5	10	8	6	4
$(4,10):$	10	4	6	5	4	3	$(5,10):$	10	5	5	4	3	2

Suppose users (1,10) and (4,2) are the two colluders, and assume the feedback sequence is $F = (10, 4, 9, 1, 7, 5)$. We expect to identify the first colluder after observing $p^2 + 1 = 5$ elements, and the second colluder after observing the next element (feedback sequence of length $p^2 + p = p(p + 1) = 6$.) That is after observing 5 elements, there is exactly one user who matches the feedback sequence in $p+1 = 3$ positions and all other possible traitors match it in at most $p = 2$ positions. By observing the 6^{th} element we can find a second colluder that matches 3 times. The following table lists (column) all possible traitors for each element of the feedback sequence. After observing 5 elements of the feedback sequence one of the colluders, (1,10) in this case, will be detected and disconnected. At this stage, the second colluder, (4,2) cannot be identified as he has appeared only twice which is the same number as the innocent user, (4,1). However by observing the 6^{th} element of the feedback sequence this colluder can also be identified.

$$\begin{array}{cccccc}
10 & 4 & 9 & 1 & 7 & 5 \\
\downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
(1,10) & (4,1) & (1,10) & (1,4) & (1,10) & (1,1) \\
(2,10) & (4,2) & (2,3) & (2,3) & (2,8) & (2,10) \\
(3,10) & (4,3) & (3,5) & (3,9) & (3,3) & (3,7) \\
(4,10) & (4,4) & (4,4) & (4,2) & (4,1) & (4,2) \\
(5,10) & (4,5) & (5,7) & (5,8) & (5,5) & (5,3) \\
 & (4,6) & & & & \\
 & (4,7) & & & & \\
 & (4,8) & & & & \\
 & (4,9) & & & & \\
 & (4,10) & & & &
\end{array}$$