Fighting two pirates

Hans Georg Schaathun

Departement of Informatics, University of Bergen, N-5020 Bergen, Norway,
georg@ii.uib.no,
WWW home page: http://www.ii.uib.no/~georg/

Abstract A pirate is a person who buys a legal copy of a copyrighted work and who reproduces it to sell illegal copies. Artists and authors are worried as they do not get the income which is legally theirs. It has been suggested to mark every copy sold with a unique fingerprint, so that any unauthorised copy may be traced back to the source and the pirate who bought it. The fingerprint must be embedded in such a way that it cannot be destroyed. Two pirates who cooperate, can compare their copies and they will find some bits which differ. These bits must be part of the fingerprint, and when the pirates can see and change these bits, they get an illegal copy with neither of their fingerprints. Collusion-secure fingerprinting schemes are designed to trace at least one of the pirates in such a collusion.

In this paper we prove that socalled $(2,2)$-separating codes often are collusion-secure against two pirates. In particular, we consider the best known explicit asymptotic construction of such codes, and prove that it is collusion-secure with better rate than any previously known schemes.

1 Fingerprinting

Once upon a time, when computing logarithms was time consuming and tables of logarithms expensive, publishers found they had to protect the tables against illegal copying. They introduced tiny errors in the least significant digits to make every copy of the tables unique. In this way an illegal copy could be traced back to one legal original, and the customer who had bought this copy could be prosecuted.

This technique, known as fingerprinting, has been suggested for digital data. Research is going on within several fields to solve the various challenges involved. One problem is the embedding. How can the copy be marked without distorting the data, and without the users being able to change the fingerprint? We are not going to address this problem any further, just state some assumptions about its solution. The problem we are going to address is how to make the system resistant against a coalition of pirates.

A vendor holds the copyright to some work, it may be a sound recording, a digital image, a literary text, or something else. A copy is a digital file which resembles the work and has the same practical (and artistic) value. A user is a legal owner of a copy, presumably bought from the vendor. A pirate is a user who makes and distributes illegal copies of the work.

A fingerprint is a word (tuple) of symbols uniquely identifying a user. The set of fingerprints is called an (n, M) code C, where n is the length of each word and M is the

number of users (or fingerprints). We let d and m denote respectively the minimum and maximum Hamming distance between two codewords, and $\delta = d/n$ is the normalised minimum distance.

The fingerprint is supposed to be embedded into the copy in order to identify its owner. The embedding of one symbol of the fingerprint is called a mark. We assume that a user investigating a single copy is unable to locate or identify any mark, and therefore cannot change any mark. A coalition of users however, can compare their copies, and any difference between their copies must be a mark. The pirates can produce copies with a false fingerprint, but every mark has to match at least one of the legal copies held by the pirates. This is known as the marking assumption.

Let P be a coalition of pirates. Since each pirate is associated to a fingerprint, we write $P \subset C$. A position i is undetectable for P if all the elements of P match in position i. The feasible set $F(P)$ is the set of false fingerprint which may be produced by P, in other words

$$F(P) = \{(c_1, \ldots, c_n) : \forall i = 1, \ldots, n, \exists (x_1, \ldots, x_n) \in P, \text{st. } c_i = x_i\}.$$

Note that $P \subset F(P)$. The elements of $F(P)$ are often called descendants of P.

The fingerprinting code C is assumed to be publicly known, however, the vendor uses a secret code C' chosen uniformly at random from the ensemble of codes equivalent to C. The codeword embedded in the copy is the codeword from C' corresponding to the codeword from C associated with the user. In this way, it is impossible for the pirates two know which coordinate position is corresponding to a given detectable mark, and which code symbol corresponds to a given value of the mark.

2 The identifiable parent property

The goal in collusion-secure fingerprinting is to identify at least one pirate when discovering a false fingerprint produced by a coalition of at most t pirates. If this is possible, we say that the code has the t-identifiable parent property (t-IPP).

Let $\mathcal{P}_t(C)$ be the family of sets $P \subset C$ of cardinality at most t. Let $P_t(\mathbf{x}) \subset \mathcal{P}_t(C)$ be the family of coalitions which could have produced $\mathbf{x}$, i.e.

$$P_t(\mathbf{x}) = \{P \in \mathcal{P}_t(C) : \mathbf{x} \in F(P)\}.$$

If the elements of $P_t(\mathbf{x})$ has a non-empty intersection for any $\mathbf{x}$, then C is t-IPP. The following definition is equivalent, and standard.

Definition 1 (Identifiable parent property). *A code C is said to have the t-identifiable parent property (t-IPP) if there is an algorithm A such that for every $P \in \mathcal{P}_t(C)$ and every vector $\mathbf{x} \in F(P)$, $A(\mathbf{x})$ returns a member of P.*

The algorithmic issues are beyond the scope of this paper. As far as we are conserned, the algorithm A may be an exhaustive search through $\mathcal{P}_t(C)$.

Observe that t-IPP implies $(t-1)$-IPP. It is well-known that binary codes cannot be even 2-IPP [BS98]. More generally the following proposition is well known in the fingerprinting literature.

Proposition 1. *Let t and q be integers. If $t < q$, then there exist asymptotic families of q-ary codes with t-IPP. If $t \geq q$, there exists no q-ary code with t-IPP and more than t codewords.*

In order to get collusion-secure codes with more pirates, we use probabilistic fingerprinting schemes. We allow A to have a certain error probability ϵ. There are two types of error: we call it a *failure* if A returns void and a *mistake* if A returns a word which is not a member of P. Mistakes is a threat to justice, as it causes innocent users to be accused. If there is no probability of mistakes, the output of A is always reliable, but occasionally there is no output at all.

A t-IPP code with ϵ-error (or (t, ϵ)-IPP) is defined as a code where the probability of error is ϵ. Failures may be turned into mistakes by picking a random codeword whenever a failure should occur, and thus past literature rarely distinguish between failure and mistake.

We define (t, ϵ)-UPP (undisputable parent property) to be a code where the algorithm A has no risk of mistake and a probability ϵ of failure. Obviously, (t, ϵ)-UPP is stronger than (t, ϵ)-IPP.

We say that a word $\mathbf{x}$ is t-identifiable if it can be traced back to one undisputable parent, that is if

$$\bigcap_{P \in P_t(\mathbf{x})} P \neq \emptyset.$$

The set of t-identifiable words is denoted $\mathcal{I}_t(C)$.

When the pirates compare their copies, they find d' detectable bits. This number d' is the Hamming distance between their two fingerprints. As long as the embedding is kept secret by the vendor, it is impossible for the pirates to tell which detected mark corresponds to which coordinate position in the code.

When they construct a false fingerprint $\mathbf{x}$, they can only choose the distance $s = d(\mathbf{a}, \mathbf{x})$. Clearly $d(\mathbf{b}, \mathbf{x}) = d' - d(\mathbf{a}, \mathbf{x})$. The two pirates cannot be distinguished, so we can assume that $s \leq d'/2$. We call s the pirate strategy, and define $\sigma := s/n$ to be the normalised strategy, where n is the length of a fingerprint.

Once the strategy is chosen, a fingerprint is produced uniformly at random from a set of $2\binom{s}{d'}$ feasible words. (If d' is even and $s = d'/2$, there are $\binom{s}{d}$ feasible words.) There is a certain probability $p_s(P)$ that the produced false fingerprint is identifiable. The pirates will obviously choose s to minimise $p_s(P)$. The probability that the pirates gets away with their forgery is $1 - p(P)$, where $p(P) = \min_s p_s(P)$.

For simplicity, we assume that the pirates know which two codewords they posess. This allows them to make a perfect minimisation of $p_s(P)$, which might not be possible in reality. Hence the $p(P)$ defined here is a lower bound on the true probability.

Theorem 1. *A q-ary code cannot have (t, ϵ)-UPP for any $t > q$ and $\epsilon < 1$.*

Proof. Consider a code and a coalition of $q + 1$ pirates. For each coordinate position, there is at least one symbol which appears in at least two of the pirate codewords. Thus the pirates has a feasible word which matches at least two pirates in each coordinate position. Since this false fingerprint is feasible for any subset of q pirates, none of the pirates are undisputable parents.

Conjecture 1. There is an asymptotic family of q-ary codes with non-zero rate and (q, ϵ)-UPP where ϵ tends to zero.

3 Separating codes

Much of the fingerprinting literature has focused on properties which are related to, but weaker than, t-IPP. The most important one of these properties is (t, t')-separation. Resently it was proved that (t, t)-separating codes can be used for constructing (t, ϵ)-IPP codes [BBK01]. We shall see that some good $(2, 2)$-separating codes are actually good $(2, \epsilon)$-IPP codes in themselves with better rates than the codes from [BBK01].

Definition 2 (Separating code). *Let $\mathbf{t} = (t_1, \ldots, t_z)$ be a tuple of natural numbers. A sequence $(T_1, \ldots, T_z)$ of pairwise disjoint vector sets is called a* **t***-configuration if $\#T_j = t_j$ for all j. Such a configuration is separated if there is a position i, such that for all $l \neq l'$ every vector of T_l is different from every vector of $T_{l'}$ on position i.*

A code is **t***-separating (a* **t***-SS) if every* **t***-configuration is separated.*

If $t_i = 1$ for all i, then **t**-separation is equivalent to z-hashing. For $z = 2$ there is a vast literature, in particular on $(2, 1)$- and $(2, 2)$-SS, it dates back at least to '69 [FGU69]. See [Sag94] for a survey.

If a code C is not $(t, 1)$-separating, there is a pirate coalition T_1 of t users who are able to forge a fingerprint $\mathbf{x}$ which belongs to a user not member of T_1. To see this, just let $(T_1, \{\mathbf{x}\})$ be a $(t, 1)$-configuration which is not separated. We say that $\mathbf{x}$ is framed by T_1, and $(t, 1)$-SS are often called t-frameproof codes in the fingerprinting literature.

If a code is (t, t)-separating, it means in fingerprinting terms, that two disjoint coalitions $T_1, T_2 \in \mathcal{P}_t(C)$ cannot produce the same false fingerprint, i.e. $F(T_1) \cap F(T_2) = \emptyset$. These codes where called t-secure frameproof in some early fingerprinting literature.

Definition 3 (Separating weights). *Let $(T_1, \ldots, T_z)$ be a* **t***-configuration. The separating weight $\theta_{\mathbf{t}}(T_1; \ldots; T_z)$ is the number of positions where the configuration is separated.*

If C is an (n, M) code, its minimum separating weight $\theta_{\mathbf{t}}$ is the least separating weight for any **t***-configuration from C. The normalised separating weight is $\tau_{\mathbf{t}} := \theta_{\mathbf{t}}/n$.*

Obviously, a code is **t**-separating if and only if $\theta_{\mathbf{t}} > 0$.

Proposition 2. *For a binary code, we have $\theta_{2,1} \geq d - m/2$.*

This result was found by Sagalovich [Sag65], but we include a proof for the reader's convenience.

Proof. Let $(\mathbf{c}', \mathbf{c}, \mathbf{a})$ be any three codewords. Since separating weights are invariant over the ensemble of equivalent codes, we can by translation assume that $\mathbf{c}' = \mathbf{0}$. We shall find a lower bound on the separating weights $\theta(\mathbf{0}, \mathbf{c}; \mathbf{a})$.

First we take $(2, 1)$-separation. Let $\mathbf{0}$, $\mathbf{c}$, and $\mathbf{a}$ be rows of a matrix. There are three types of columns; Type R is $(001)^{\mathrm{T}}$ which are the ones giving separation, Type 0 is $(000)^{\mathrm{T}}$, and Type I is $(010)^{\mathrm{T}}$ and $(011)^{\mathrm{T}}$. Let v_i be the number of columns of Type i. We have that $\theta(\mathbf{0}, \mathbf{c}; \mathbf{a}) = v_{\mathrm{R}}$ and $w(\mathbf{c}) = v_{\mathrm{I}}$.

Define

$$\Sigma := d(\mathbf{0}, \mathbf{a}) + d(\mathbf{c}, \mathbf{a}) = 2v_{\mathrm{R}} + v_{\mathrm{I}} = 2\theta(\mathbf{0}, \mathbf{c}; \mathbf{a}) + w(\mathbf{c}).$$

Since Σ is the sum of two distances, we have

$$2d \leq \Sigma \leq 2m,$$

so

$$2\theta(\mathbf{0}, \mathbf{c}; \mathbf{a}) = \Sigma - w(\mathbf{c}) \geq 2d - m.$$

It has also be shown that $\theta_{2,2} \geq 2d - 3m/2$ in a similar way. A corollary is that if $\delta > \frac{3}{4}$, then $\theta_{2,2}$ is non-zero, and the code is $(2, 2)$-separating.

Proposition 3. *Let* $\mathbf{t}$ *be a tuple of natural numbers. If* C_1 *is a* M'*-ary* $[n_1, M]$ *code with separating weight* $\theta'_{\mathbf{t}}$*, and* C_2 *is a* q*-ary* $[n_2, M']$ *code with separating weight* $\theta''_{\mathbf{t}}$*, then the concatenation* C *of the two codes is a* $[n, M]$ *code with* $n = n_1 n_2$ *and separating weight* $\theta_{\mathbf{t}} \geq \theta'_{\mathbf{t}} \theta''_{\mathbf{t}}$*.*

Proof. Let $(T_1, \ldots, T_z)$ be any $\mathbf{t}$-configuration from C. Then there is a corresponding $\mathbf{t}$-configuration $(T'_1, \ldots, T'_z)$ in C_1, which is separated in at least $\theta'_{\mathbf{t}}$ positions.

Now consider a single position i, where $(T'_1, \ldots, T'_z)$ is separated. Each symbol in this position corresponds to a word in C_2, so $(T'_1, \ldots, T'_z)$ corresponds to a collection of subsets $(T''_1, \ldots, T''_z)$ in C_2. Since $(T'_1, \ldots, T'_z)$ is a separated $\mathbf{t}$-configuration, $(T''_1, \ldots, T''_z)$ must also be a separated $\mathbf{t}$-configuration, and since C_2 has separating weight $\theta''_{\mathbf{t}}$, it follows that $(T''_1, \ldots, T''_z)$ is separated in at least $\theta''_{\mathbf{t}}$ positions.

We conclude that $(T_1, \ldots, T_z)$ is separated in at least $\theta'_{\mathbf{t}} \theta''_{\mathbf{t}}$ positions, and since this holds for any $\mathbf{t}$-configuration, the proposition follows.

Corollary 1. *The concatenation of two* $\mathbf{t}$*-SS is a* $\mathbf{t}$*-SS.*

The current best constructible rate for asymptotic $(2, 2)$-SS is 0.026. This was constructed in [CELS01] by concatenating an asymptotic code with $\delta > \frac{3}{4}$ with a small inner code which had been explicitely confirmed to be $(2, 2)$-separating. However, Sagalovich [Sag94] had already given a different construction of $(2, 2)$-SS with this rate.

The outer code used in the construction is one due to Tsfasman. He showed in [Tsf91], that there is an asymptotic class of q-ary codes with rate R and minimum distance δ whenever

$$R + \delta < 1 - (\sqrt{q} - 1)^{-1}.$$

The inner code is the punctured dual C' of a two-error-correcting BCH code with parameters $[126, 14, 55]$. This code was proven to be 3-wise intersecting in [CZ94], a property which is equivalent to $(2, 2)$-separation [BR80]. To see that C' is $(2, 2)$-separating, we recall that the dual of 2-BCH has only two weights, $2^{2t} - 2^t$ and $2^{2t} + 2^t$. Consequently C' has $d \geq 2^{2t} - 2^t - 1$ and $m \leq 2^{2t} + 2^t$, and

$$4d - 3m \geq 2^{2t} - 7 \cdot 2^t - 4,$$

which is greater than zero whenever $t \geq 3$. Our code C', has $m = 72$, so $\theta_{2,1} \geq 55 - 72/2 = 19$.

The specific outer code shall have $q = 2^{14}$, and since we require $\delta \approx 0.75$, we get $R \approx 1 - 127^{-1} - 0.75 \approx 0.2421$. The $(2,1)$-separating weight is $\tau_{2,1} \geq 0.75 - 0.5 = 0.25$. The concatenated code $\mathfrak{C}$ will have $R \approx 0.026$, $\delta \approx 0.3274$, and $\tau_{2,1} \geq 0.03770$. As mentioned, this construction is not new. The new result, which will be proved in the next section, is that C is $(2,\epsilon)$-UPP where ϵ tends to 0 with increasing n.

4 Binary separating codes for fingerprinting

In the sequel, we assume a binary code. Let $m(\mathbf{a},\mathbf{b},\mathbf{c})$ be the word obtained by majority voting of the three vectors $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$. That is, in each position i in $m(\mathbf{a},\mathbf{b},\mathbf{c})$ contains the symbol which occurs in position i of at least two of the three vectors $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$.

Lemma 1. *If C is an (n, M) $(2,2)$-SS, then for any $P = \{\mathbf{a},\mathbf{b}\} \subset C$, we have*

$$F(P)\backslash\mathcal{I}_2(C) = \{m(\mathbf{a},\mathbf{b},\mathbf{c}) : \mathbf{c} \in C\backslash P\},$$

and

$$\#(F(P)\backslash\mathcal{I}_2(C)) = M - 2.$$

This result was pointed out in [Löf01].

Proof. Let $\mathbf{x}$ be a vector which is not identifiable. Because C is $(2,2)$-separating, the possible parent sets of $\mathbf{x}$ must form a triangle, i.e. $\{\mathbf{a},\mathbf{b}\}$, $\{\mathbf{a},\mathbf{c}\}$, and $\{\mathbf{b},\mathbf{c}\}$. The only vector $\mathbf{x}$ which is feasible for any of the three sets is $m(\mathbf{a},\mathbf{b},\mathbf{c})$.

Given a pirate coalition $P = \{\mathbf{a},\mathbf{b}\}$, there are $M-2$ possible triangles, for $\mathbf{c} \in C\backslash P$. For each triangle there is one word which is not identifiable.

Lemma 2. *For any strategy $s < \theta_{2,1}$ we get $p_s(P) = 1$.*

Proof. Let $\mathbf{a}$ and $\mathbf{b}$ be a pirate. If the pirates manage to forge a fingerprint $\mathbf{x}$ forming a triangle with a third codeword $\mathbf{c}$, then $\mathbf{c}$ must match $\mathbf{a}$ in s out of the $d(\mathbf{a},\mathbf{b})$ detectible marks and match $\mathbf{b}$ on the others. These positions are exactly the ones where $(P,\{\mathbf{c}\})$ is separated, so if $s < \theta_{2,1}$, then no such $\mathbf{c}$ can exist.

Theorem 2. *Let C be a $(2,2)$-SS, and let $P \subset C$ be any pirate coalition of size at most $t = 2$. For any strategy s, the probability that P escapes detection is*

$$1 - p_s(P) \leq (M-2)\min\left\{\frac{1}{2}\binom{n\delta}{n\tau_{2,1}}^{-1}, \binom{n\delta}{n\delta/2}^{-1}\right\}.$$

Proof. Let $\kappa(s)$ be the probability of chosing a particular fingerprint given a strategy s. By Lemma 2, we can assume $s \geq \theta_{2,1}$. We have

$$\kappa(d(\mathbf{a},\mathbf{b})/2) = \binom{d(\mathbf{a},\mathbf{b})}{d(\mathbf{a},\mathbf{b})/2}^{-1} \leq \binom{n\delta}{n\delta/2}^{-1}.$$

If $s \neq d(\mathbf{a},\mathbf{b})/2$, we get

$$\kappa(s) = \frac{1}{2}\binom{d(\mathbf{a},\mathbf{b})}{s}^{-1} \leq \frac{1}{2}\binom{n\delta}{n\tau_{2,1}}^{-1},$$

The number of non-identifiable words is $\mu = M - 2$, so there cannot be more than μ feasible false fingerprints allowing the pirates to escape. Multiplying μ with $\kappa(s)$ we get the theorem.

If we take assymptotic values for increasing n, we arrive at the following corollary, where H is the natural entropy function.

Corollary 2. *Any* $(2,2)$*-SS is a* $(2,\epsilon)$*-UPP with*

$$\epsilon \leq e^{\lambda n},$$

where

$$\lambda = R\ln 2 - H(\tau_{2,1}/\delta)\delta.$$

Considering our $(2,2)$-SS $\mathfrak{C}$ with $R \approx 0.026$ and $\tau_{2,1} \geq 0.03770$, we get the following values:

$$\lambda \approx -0.09891,$$
$$\epsilon \leq 0.9058^n,$$

which leads to the following theorem.

Theorem 3. *There is a constructible asymptotic binary code with* $(2,\epsilon)$*-UPP with rate* $R \approx 0.026$ *and failure rate* $\epsilon \leq 0.9058^n$.

The code $\mathfrak{C}$ has better rate than any $(2,\epsilon)$-IPP known from past literature. Though the code has been known, it is a new result that it has UPP, or even IPP. Unfortunately, the results does not extend very well, since Theorem 1 rules out any q-ary (t,ϵ)-UPP codes for $t > q$.

5 Discussion

We have introduced a probabilistic 2-IPP code with a rate better than anything we have managed to locate in the literature. Furthermore, with this code, there is no risk of accusing an innocent user. It still remains to construct an efficient tracing algorithm usable with the present code.

We have seen that codes with (t,ϵ)-UPP cannot exist for $t > q$, but it is an open question whether the present techniques can be modified to construct (t,ϵ)-IPP codes with $t > q$. It would also be interesting to construct q-ary codes with (q,ϵ)-UPP for arbitrary q.

References

BBK01. A. Barg, G. R. Blakley, and G. Kabatiansky. Good digital fingerprinting codes. Technical report, DIMACS, 2001.

BR80. Bella Bose and T. R. N. Rao. Separating and completely separating systems and linear codes. *IEEE Trans. Comput.*, 29(7):665–668, 1980.

BS98. Dan Boneh and James Shaw. Collusion-secure fingerprinting for digital data. *IEEE Trans. Inform. Theory*, 44(5):1897–1905, 1998. Presented in part 1995, see Springer LNCS.

CELS01. Gérard D. Cohen, Sylvia B. Encheva, Simon Litsyn, and Hans Georg Schaathun. Intersecting codes and separating codes. *Discrete Applied Mathematics*, 2001. To appear.

CZ94. Gérard Cohen and Gilles Zémor. Intersecting codes and independent families. *IEEE Trans. Inform. Theory*, 40:1872–1881, 1994.

FGU69. A. D. Friedman, R. L. Graham, and J. D. Ullman. Universal single transition time asynchronous state assignments. *IEEE Trans. Comput.*, 18:541–547, 1969.

Löf01. Jacob Löfvenberg. *Codes for Digital Fingerprinting*. PhD thesis, Linköpings Universitet, 2001. `http://www.it.isy.liu.se/publikationer/index.html`.

Sag65. Yu. L. Sagalovich. A method for increasing the reliability of finite automata. *Problems of Information Transmition*, 1(2):27–35, 1965.

Sag94. Yu. L. Sagalovich. Separating systems. *Problems of Information Transmition*, 30(2):105–123, 1994.

Tsf91. Michael A. Tsfasman. Algebraic-geometric codes and asymptotic problems. *Discrete Appl. Math.*, 33(1-3):241–256, 1991. Applied algebra, algebraic algorithms, and error-correcting codes (Toulouse, 1989).