On the IPP Properties of Reed-Solomon Codes

Marcel Fernandez, Josep Cotrina, Miguel Soriano, and Neus Domingo *

Abstract Codes with traceability properties are used in schemes where the identification of users that illegally redistribute content is required. For any code with traceability properties, the Identifiable Parent Property (c-IPP) seems to be less restrictive than the Traceability (c-TA) property. In this paper, we show that for Reed-Solomon codes both properties are in many cases equivalent. More precisely, we show that for an $[n,k,d]$ Reed-Solomon code, defined over a field that contains the $n-d$ roots of unity, both properties are equivalent. This answers a question posted by Silverberg et al. in [10, 11], for a large family of Reed-Solomon codes.

1 Introduction

The concept of traitor tracing was coined in [5] as a method to discourage piracy. Traitor tracing schemes are useful in scenarios where the distributed content may only be accessible to authorized users, like decrypting broadcast messages, software installation and distribution of multimedia content.

This paper discusses the characteristics of the *identifiable parent property* (IPP) of Reed-Solomon codes used in traitor tracing and fingerprinting schemes. However, before we get into technical matters, we give an intuitive overview. By doing this

Marcel Fernandez · Josep Cotrina
Technical University of Catalonia, Department of Telematic Engineering

Miguel Soriano
CTTC. Centre Tecnologic de Telecomunicacions de Catalunya, Barcelona (Spain)

Neus Domingo
I.E.S. J.V Foix. Rubi. Barcelona (Spain)

* This work has been supported in part by the Spanish Research Council (CICYT) Project TSI2005-07293-C02-01 (SECONNET), by CICYT Project TEC2006-04504 and by CONSOLIDER CSD2007-00004 “ARES”, funded by the Spanish Ministry of Science and Education.

at the beginning of the paper, we try to separate the concepts from where our work emanates from the intrinsic mathematical development and also hopefully provide the reader an extra motivation for going deep into our results.

The scenario we will deal with is the following one. A distributor D, that sells digital content, wishes to discourage illegal redistribution of its products. To this end, he embeds a *unique* set of symbols to each copy of the content before it is delivered. This makes each copy unique and therefore if a dishonest user illegally redistributes his copy, he can be unambiguously identified by simply extracting the set of symbols.

A weakness to this scheme can be spotted by noting that a coalition of two or more dishonest users can get together and by comparing their copies they perform a *collusion attack*. This attack consists in detecting the positions in which their copies differ and with this knowledge, they create a new copy that in every detected position contains a symbol of one of the members of the coalition. This new copy is a pirate copy that tries to disguise the identity of the guilty users and is the one they redistribute.

More precisely, the distributor assigns a codeword from a q-ary fingerprinting code to each user. To embed the codeword into each users object, the object is first divided into blocks. The distributor then picks a set of these blocks at random. This set of blocks is kept secret and will be the same for all users. Then using a watermarking algorithm a mark of the fingerprint codeword is embedded in each block. Note that a given user will have one of the q versions of the block. The colluding traitors compare their copies, detect the blocks where their copies differ and with this information at hand, they construct a pirate copy where each block belongs to the corresponding block of one of the traitors. Since each mark is embedded using a different random sequence, and these sequences are unknown to the traitors, they cannot create a version of the block that they do not have.

With the above scenario in mind, it is clear that the distributor D, has to embed sets of symbols that are secure against collusion attacks. One way to obtain such sets is by using codes with the *Identifiable Parent Property* (c-IPP).

1.1 Previous work

Codes with the IPP were introduced in [8]. Informally, and using the traitor tracing scenario described above, a code has the c-IPP property if given a pirate copy, all coalitions of at most c traitors that can generate this pirate copy have a non-empty intersection.

The IPP has received considerable attention in the recent years, having been studied by several authors [3, 4, 13, 9, 14, 1, 2, 7].

A stronger property is the Traceability (c-TA) property. In this case given a pirate copy, one of the traitors involved in its creation is the closest one in terms of the Hamming metric.

In [12], sufficient conditions for a linear error correcting code to be a c-TA code are given. Efficient algorithms for the identification of traitors in schemes using c-TA codes are discussed in [10, 11].

In [10, 11] it is stated that tracing for TA codes is an $O(N)$ process, with N the number of users, whereas for IPP codes tracing is more expensive since it is an $O(\binom{N}{c})$ process. Since the TA property is stronger than the IPP, and tracing is far more expensive for the IPP, it seems natural to expect that by relaxing the TA requirements one could still have a code that, even though in no longer c-TA, still possesses IPP. However in [11] some examples using truncated Reed-Solomon codes lead toward the opposite, that is, if a Reed-Solomon code does not have the TA property then it does not have the IPP one either.

1.2 Our contribution

In this paper we answer a question posted by Silverberg et al. in [10, 11]. The results we present hopefully give way to a total understanding of the IPP property in Reed-Solomon codes.

In [12, Lemma 1.3] authors prove that a c-TA code is a c-IPP code. However as seen before, the TA property is stronger than IPP, taking this into account Silverberg et al. in [10, 11] asked the following question:

Question 11 [11]: It is the case that all c-IPP Reed-Solomon codes are c-TA codes?

Below, and as a result of expressing the IPP in an algebraic manner, we give an affirmative answer to this question for a large family of Reed-Solomon codes. Surprisingly enough, the answer is positive for codes defined over a field that contains the $n-d$ roots of unity. Note that our results imply that for this family of Reed-Solomon codes, failing to be c-TA also involves failing to be c-IPP.

For a more precise statement of the Question 11 [11], see Section 2.1 below.

1.3 Organization of the paper

The paper is organized as follows. In Section 2, we provide the necessary background in coding theory, traceability and IPP. In Section 3 we start our discussion by defining a set of polynomials that allow us to express the IPP algebraically. The main result of the paper is presented in Section 4, and comes in the form of a theorem giving the necessary and sufficient conditions for Reed-Solomon codes to be c-IPP codes. A complete example to clarify our results is given in Section 5. We draw our conclusions in Section 6.

2 Definitions and previous results

We define a *code* as a set of n-tuples of elements from a set of scalars. The set of scalars is called the *code alphabet*. An n-tuple in the code is called a *word* and the elements of the code are called *code words*. If the code alphabet is a finite field $\mathbb{F}_q$, then a code C is a *linear code* if it forms a vectorial subspace. The dimension of the code is defined as the dimension of the vectorial subspace.

Let $\mathbf{a},\mathbf{b} \in \mathbb{F}_q^n$ be two words, then the *Hamming distance* $d(\mathbf{a},\mathbf{b})$ between $\mathbf{a}$ and $\mathbf{b}$ is the number of positions where $\mathbf{a}$ and $\mathbf{b}$ differ. Let C be a code, the *minimum distance* of C, $d(C)$, is defined as the smallest distance between two different codewords.

A linear code with length n, dimension k and minimum distance d is denoted as a $[n,k,d]$-code, or simply as an (n,d) code.

A well known class of linear codes are Reed-Solomon codes, that can be defined as follows:

Let $\mathbb{F}_q[x]$ be the ring of polynomials defined over $\mathbb{F}_q$. Consider the set of polynomials of degree less than k, $\mathbb{F}_q[x]_k \subset \mathbb{F}_q[x]$. Let γ be a primitive element of $\mathbb{F}_q$, and $\lambda_1,\ldots,\lambda_n \in \mathbb{F}_q - \{0\}$.

Definition 1 *We define a generalized Reed-Solomon, $RS[n,k]_q$, code as the vectorial subspace of $\mathbb{F}_q^n$ determined by the vectors of the form*

$$\mathbf{v} = (\lambda_1 f(\gamma^1),\ldots,\lambda_n f(\gamma^{q-1}))$$

where $f \in \mathbb{F}_q[x]_k$. Note that $n = q-1$.

2.1 Background and previous results on c-IPP traceability codes

Given a code $C(n,d)$ defined over the finite field of q elements, $\mathbb{F}_q$, where n denotes the code length and d the minimum distance of the code, the *set of descendants* (false fingerprint) of any subset $T = \{\mathbf{t}^1,\ldots,\mathbf{t}^c\} \subseteq C$, where $\mathbf{t}^i = (t_1^i,\ldots,t_n^i)$, denoted $desc(T)$, is defined as

$$desc(T) = \left\{ \mathbf{y} = (y_1,\ldots,y_n) \in \mathbb{F}_q^n | y_i \in \{t_i^j | \mathbf{t}^j \in T\}, 1 \le i \le n \right\}.$$

Definition 2 *A code C is a c-traceability code (denoted c-TA), for $c > 0$, if for all subsets (coalitions) $T \subseteq C$ of at most c code words, if $\mathbf{y} \in desc(T)$, then there exists a $\mathbf{t} \in T$ such that $d(\mathbf{y},\mathbf{t}) < d(\mathbf{y},\mathbf{w})$ for all $\mathbf{w} \in C - T$.*

Definition 3 *A code $C(n,d)$, defined over $\mathbb{F}_q$, is a c-identifiable parent property code (denoted c-IPP), $c > 0$, if for all $\mathbf{y} \in \mathbb{F}_q^n$ and all the coalitions $T \subseteq C$ of at most c code words, we have $\mathbf{y} \notin \bigcup_T desc(T)$ or*

$$\bigcap_{\mathbf{y}\in desc(T)} T \neq \emptyset.$$

In [12, Lemma 1.3] it is shown that that a c-TA code is a c-IPP code. In [5][6][12, Theorem 4.4] it is proved that any $C(n,d)$ code with $d > n - n/c^2$ is a c-TA code. Moreover, if $C(n,d)$ is a code defined over $\mathbb{F}_q$, in [12, Lemma 1.6] authors show that if $|C| > c \geq q$ then C is not a c-IPP code.

Given a code $C(n,d)$, authors in [11, Section IV], construct unordered sets from the ordered sets that constitute the code as follows: to a codeword $\mathbf{x} = (x_1,\ldots,x_n) \in C$ they associate the set $x' = \{(1,x_1),\ldots,(n,x_n)\}$. Then they define TA set systems (as opposed to TA codes) in the natural way, with the noteworthy difference that a pirate unordered set (unordered fingerprint) consist of n elements such that each element is a member of some coalition member's set. In [11, Theorem 7], authors prove that if $C(n,d)$ is a Reed-Solomon code with minimum distance $d \leq n - n/c^2$ then the set system corresponding to C is not a c-TA system. Note that this result does not implies that $d > n - n/c^2$ is a necessary condition for RS codes to be c-TA.

Moreover in [11, Theorem 8] authors construct a family of truncated ($n < q-1$) $RS[n,k]_q$ codes that fail to be c-IPP if $c^2 > n/(n-d)$.

Then in [11, Question 11] the authors ask if it is always true that the c-IPP fails if $c^2 > n/(n-d)$.

In this paper we give another partial positive answer of this question, showing that there are other families of Reed-Solomon codes that fail to be c-IPP if $c^2 > n/(n-d)$. Obviously this does not close the problem, but we think that it gives some hints that may hopefully be useful in finding the final response.

3 The IPP condition for Reed-Solomon codes

In this section we set the ground for the discussion of our main results. Informally, we define a set of polynomials (denoted $h_{ij}(x)$), that help us construct an algebraic representation of the IPP. Using these polynomials, we set up a system of equations for which the existence of a solution implies that the code is not c-IPP. In Section 4, we will show how to solve this equation system for a large number of Reed-Solomon codes.

Let $0 < c_1 \leq c_2$ be two integer numbers. We say that a code C is not a (c_1,c_2)-IPP code if there exist coalitions T_1 of c_1 code words and T_2 of c_2 codewords, such that

$$desc(T_1) \bigcap desc(T_2) \neq \emptyset \text{ and } T_1 \bigcap T_2 = \emptyset.$$

Obviously, from Definition 3 the code C is not c_2-IPP.

Therefore, if a $RS[n,k]_q$ code fails to be (c_1,c_2)-IPP this means that there exist two disjoint coalitions, with c_1 and c_2 distinct code words respectively, $T_1 = \{f_0(x),\ldots,f_{c_1-1}(x)\}$, $T_2 = \{g_0(x),\ldots g_{c_2-1}(x)\}$ (where $f_i(x), g_j(x) \in \mathbb{F}_q[x]_k$, but with an abuse of notation they also represent vectors of the form

$$\mathbf{f_i} = (\lambda_1 f_i(\gamma^1), \ldots, \lambda_n f_i(\gamma^n)$$

), with $T_1 \cap T_2 = \emptyset$ and that can generate the same descendant (false fingerprint) $\mathbf{y}$.

We can always assume that code word $\mathbf{0}$ is a code word of coalition T_1, otherwise consider coalitions $T_1 - \mathbf{f_0} = \{\mathbf{f_0} - \mathbf{f_0}, \ldots, \mathbf{f_{c_1-1}} - \mathbf{f_0}\}$ and $T_2 - \mathbf{f_0} = \{\mathbf{g_0} - \mathbf{f_0}, \ldots, \mathbf{g_{c_2-1}} - \mathbf{f_0}\}$. Then it is not difficult to verify that $(T_1 - \mathbf{f_0}) \cap (T_2 - \mathbf{f_0}) = \emptyset$ and they both can generate the fingerprint $\mathbf{y} - \mathbf{f_0}$. Thus, in what follows, we will assume that $\mathbf{f_0} = \mathbf{0}$.

We define polynomials

$$h_{ij}(x) \triangleq f_i(x) - g_j(x) = \beta_{ij} \prod_{k=1}^{s_{ij}} (x - \alpha_k^{ij}) \tag{1}$$

for $i = 0, \ldots, c_1 - 1$ and $j = 0, \ldots, c_2 - 1$.

The polynomials $h_{ij}(x)$ will be a key tool in all the subsequent work. In a sense, they allow us to have an algebraic representation of the IPP .

Note that the polynomials $h_{ij}(x)$ have at most $n - d = k - 1$ roots, thus $s_{ij} \leq n - d$, otherwise two distinct code words in the code would agree in more than $n - d$ coordinates, and this is not possible.

We will make an extensive use of the following result:

Lemma 4 *If a RS$[n,k]_q$ code fails to be (c_1, c_2)-IPP, (T_1 and T_2 can generate the same descendant), then the set of roots of the set of polynomials $\{h_{ij}(x)\}$ is $\mathbb{F}_q - \{0\}$. Therefore, $\sum_{ij} s_{ij} \geq n$, $x^n - 1 | \prod_{ij} h_{ij}(x)$ and $c_1 c_2 (n - d) \geq n$.*

Proof. The proof is straight forward from the definition of the polynomials $h_{ij}(x)$ and the definition of the (c_1, c_2)-IPP.

□

In the previous reasoning we have seen that we always can take $f_0(x) = 0$, therefore

$$g_j(x) = f_0(x) - h_{0j}(x) = -\beta_{0j} \prod_{k=1}^{s_{0j}} (x - \alpha_k^{0j}) \qquad j = 0, \ldots, c_2 - 1$$

Since $f_i(x) = \sum_{k=0}^{n-d} f_i^k x^k$ for $i = 1, \ldots, c_1 - 1$ then we can write down the following equation system(with an abuse of notation, because we are assuming that $s_{ij} = n - d$ for all i, j, that in fact is the worst case situation):

$$\left.\begin{array}{ll} f_i^0 & = \beta_{ij} \prod \left(-\alpha_k^{ij}\right) - \beta_{0j} \prod \left(-\alpha_k^{0j}\right) \\ \cdots & \\ f_i^{n-d-1} & = -\beta_{ij} \left(\sum \alpha_k^{ij}\right) + \beta_{0j} \left(\sum \alpha_k^{0j}\right) \\ f_i^{n-d} & = \beta_{ij} - \beta_{0j} \end{array}\right\} \tag{2}$$

where $i = 1, \ldots, c_1 - 1$ and $j = 0, \ldots, c_2 - 1$.

Note that if this equation system has a solution then the associated Reed-Solomon is not (c_1, c_2)-IPP.

When finding a solution for (2), we observe that the equation system has $(c_1 - 1)c_2(n-d+1)$ equations, and $(c_1-1)(n-d+1)+c_1c_2+n$, degrees of freedom. However, there is an important restriction due to Lemma 4, that is, the values of the α_k^{ij} ($i = 0,\ldots,c_1-1$ and $j = 0,\ldots,c_2-1$) must take distinct n values in $\mathbb{F}_q$, and this reduces the chance to find a solution. Note that if we assume that the values of α_k^{ij} are arbitrarily assigned then we only have $(c_1-1)(n-d+1)+c_1c_2$ degrees of freedom.

Below, in Section 4, we will show how a solution can be found for a large family of Reed-Solomon codes.

Before concluding this section, we review some trivial results on non IPP conditions.

Lemma 5 *Here we consider $[n,k,d]$ Reed-Solomon codes and assume that the code length n is fixed.*

1. *For a fixed value d, if the code is not $(c_1,c_2)-$IPP then it is not (c_1',c_2')-IPP, for any pair of values $c_1' \geq c_1, c_2' \geq c_2$.*
2. *If the code is not $(c_1,c_2)-$IPP for some value of d, then it is not (c_1,c_2)-IPP for any value $d' \leq d$.*

4 Main result on IPP Reed-Solomon codes

In this section we discuss the main result in this paper that gives an answer to the question posed in [11] ([Question 11]) asking whether if it is always true that c-IPP property fails if $c^2 > n/(n-d)$.

In Theorem 6 below, we show that in fact is true that c-IPP property fails if $c^2 > n/(n-d)$, for all Reed-Solomon codes defined over a field that contains the $n-d$ roots of unity.

Intuitively, our strategy is as follows. From Lemma 4 and the subsequent reasoning, it is clear that, if for a given code the equation system (2) has a solution then the code is not (c_1,c_2)-IPP. Since (2) has more equations (although may of the equations might be redundant) than degrees of freedom it is necessary to invert this situation. We accomplish this by finding a suitable set of polynomials $h_{ij}(x)$.

Theorem 6 *Let $RS[n,k]_q$ be a Reed-Solomon code. Consider two integer numbers $c_1 \leq c_2$. If $c_1c_2 < n/(n-d)$ the code is (c_1,c_2)-IPP. Moreover, if $n-d$ divides $q-1$, then the code is (c_1,c_2)-IPP if and only if $c_1c_2 < n/(n-d)$.*

Proof. The sufficient part it is already known, but we prove again it for completeness. If we consider a coalition T_1 of at most c_1 code words that can produce a descendant (false fingerprint) $\mathbf{y}$, we can ensure that one of the c_1 code words in the coalition agrees with $\mathbf{y}$ in at least $n/c_1 > (n-d)c_2$ of the coordinates. But any member $\mathbf{v}$ of any coalition T_2 of at most c_2 code words, with $T_2 \cap T_1 = \emptyset$, can only agree with $\mathbf{y}$ in at most $(n-d)c_1$ coordinates, otherwise $\mathbf{v}$ shall coincide in more

than $n-d$ coordinates with a code word in T_1, and this is not possible because of the definition of minimum distance of a code. Thus the code words in coalition T_2 can generate at most $(n-d)c_1c_2 < n$ coordinates of $\mathbf{y}$, that is, they can not generate the descendant $\mathbf{y}$.

For the necessary condition, in virtue of Lemma 5 we can assume that $c_1c_2 = \lceil n/(n-d) \rceil$.

If $n-d$ divides $q-1$, we have that the $(n-d)$-roots of the unity belong to $\mathbb{F}_q$. Let $s = (q-1)/(n-d)$, then we can express the $(n-d)$-roots of the unity as α^{sk}, where α is a primitive element of $\mathbb{F}_q$.

We define the polynomial

$$P(x) \triangleq \prod_{k=1}^{n-d}(x-\alpha^{ks}) = x^{n-d}-1.$$

now we can express the polynomials $h_{ij}(x)$ as

$$h_{ij}(x) \triangleq \beta_{ij}P(\alpha^{ic_2+j}x) = \beta_{ij}\alpha^{(ic_2+j)(n-d)}\prod_{k=1}^{n-d}\left(x-\alpha^{ks-ic_2-j}\right) =$$

$$= \beta_{ij}\alpha^{(ic_2+j)(n-d)}x^{n-d} - \beta_{ij},$$

for $i = 0,\ldots,c_1-1, j = 0,\ldots c_2-1$, where $c_1c_2 \geq s$. Clearly the α^{ks-ic_2-j}'s take as value all the elements in $\mathbb{F}_q - \{0\}$ for $i = 0,\ldots,c_1-1,\ j = 0,\ldots c_2-1$ and $k = 1,\ldots,n-d$.

Now, the equation system (2) can be re-expressed as

$$\left.\begin{array}{l} f_i^0 = -\beta_{ij} + \beta_{0j} \\ f_i^{n-d} = \beta_{ij}\alpha^{(ic_2+j)(n-d)} - \beta_{0j}\alpha^{j(n-d)} \end{array}\right\} \tag{3}$$

To solve this system we first will take $f_i^{n-d} = 0$ (in other words, we take the $f_i(x)$ polynomials as constant). We have that

$$\begin{array}{ll} f_i^{n-d} & = 0 \\ \beta_{ij} & = \beta_{0j}\alpha^{-ic_2(n-d)} \end{array} \tag{4}$$

and by taking $\beta_{0j} = 1$, it follows that

$$\begin{array}{ll} \beta_{0j} & = 1 \\ f_i^0 & = -\alpha^{-ic_2(n-d)} + 1 \end{array} \tag{5}$$

It is clear that (5) solves the equation system (2), and the theorem is proved. However, before we finish the proof perhaps some observations are in order.

First note that if $f_i^0 = 0$ for some i, then we would have more than a single zero polynomial, however since we are assuming that $c_1c_2 = \lceil n/(n-d) \rceil$ this can not happen.

Also, note that the equation system (2) is simplified since the coefficients of degree $s \neq 0, n-d$ of $h_{ij}(x)$ and of $g_j(x) = h_{0j}(x)$ are 0, and so the equations are

of the form "f_i^s *equals* 0", for all j, and therefore are satisfied trivially by simply taking $f_i^s = 0$.

Finally, observe that if $\alpha^{j(n-d)} = -1 = \alpha^{n/2}$ then $2c_2 > \lceil n/(n-d) \rceil$, thus $c_1 \leq 1$, and $c_2 = \lceil n/(n-d) \rceil$. But this directly implies that the code is not (c_1, c_2)-IPP.

□

5 Example

In this section we present an example of the above results.

We take a Reed-Solomon code over $\mathbb{F}_{13}$ (q=13). We denote the elements of $\mathbb{F}_{13}$ as $\{0,1,2,3,4,5,6,7,8,9,10,11,12\}$. Since we wish to prove that if $d < n - \frac{n}{c^2}$ then the code is not c-IPP then we take the code with parameters $[n = 12, k = 4, d = 9]$ and $c = 2$. Note that $n-d$ divides $q-1$ ($\mathbb{F}_{13}$ contains the $n-d = 3$ roots of unity).

With the above reasoning in mind, we need to find polynomials $f_0(x), f_1(x)$, $g_0(x), g_1(x)$ such that when grouped into two disjoint coalitions the corresponding code words can generate the same descendant (false fingerprint). In other words, we wish to find (disjoint) Coalition 1 $\{f_0(x) = 0, f_1(x)\}$ and Coalition 2 $\{g_0(x), g_1(x)\}$ such that their corresponding code words $\{\mathbf{f_0}, \mathbf{f_1}\}$ and $\{\mathbf{g_0}, \mathbf{g_1}\}$ can generate the same exact descendant (false fingerprint).

First of all, we define the h_{ij} polynomials.

$$\begin{array}{l} h_{00} = f_0 - g_0 = \beta_{00}(x-\alpha_1^{00})(x-\alpha_2^{00})(x-\alpha_3^{00}) \\ h_{01} = f_0 - g_1 = \beta_{01}(x-\alpha_1^{01})(x-\alpha_2^{01})(x-\alpha_3^{01}) \\ h_{10} = f_1 - g_0 = \beta_{10}(x-\alpha_1^{10})(x-\alpha_2^{10})(x-\alpha_3^{10}) \\ h_{11} = f_1 - g_1 = \beta_{11}(x-\alpha_1^{11})(x-\alpha_2^{11})(x-\alpha_3^{11}) \end{array} \tag{6}$$

where the α_l^{ij} take *all* of the non-zero values of $\mathbb{F}_{13}$.

Taking into account that $f_0(x) = 0$, we have that:

$$\begin{array}{rl} g_0 = -h_{00} & = -\beta_{00}(x-\alpha_1^{00})(x-\alpha_2^{00})(x-\alpha_3^{00}) \\ g_1 = -h_{01} & = -\beta_{01}(x-\alpha_1^{01})(x-\alpha_2^{01})(x-\alpha_3^{01}) \\ f_1 = h_{10} + g_0 & = \beta_{10}(x-\alpha_1^{10})(x-\alpha_2^{10})(x-\alpha_3^{10}) \\ & \quad - \beta_{00}(x-\alpha_1^{00})(x-\alpha_2^{00})(x-\alpha_3^{00}) \\ f_1 = h_{11} + g_1 & = \beta_{11}(x-\alpha_1^{11})(x-\alpha_2^{11})(x-\alpha_3^{11}) \\ & \quad - \beta_{01}(x-\alpha_1^{01})(x-\alpha_2^{01})(x-\alpha_3^{01}) \end{array} \tag{7}$$

and since $f_1(x) = f_1^0 + f_1^1 x + f_1^2 x^2 + f_1^3 x^3$ (because $k-1 = n-d = 3$), it follows that the system to be solved is

$$\left.\begin{array}{l} f_1^0 = -\beta_{10}\alpha_1^{10}\alpha_2^{10}\alpha_3^{10} + \beta_{00}\alpha_1^{00}\alpha_2^{00}\alpha_3^{00} \\ f_1^1 = \beta_{10}(\alpha_1^{10}\alpha_2^{10} + \alpha_1^{10}\alpha_3^{10} + \alpha_2^{10}\alpha_3^{10}) - \beta_{00}(\alpha_1^{00}\alpha_2^{00} + \alpha_1^{00}\alpha_3^{00} + \alpha_2^{00}\alpha_3^{00}) \\ f_1^2 = -\beta_{10}(\alpha_1^{10} + \alpha_2^{10} + \alpha_3^{10}) + \beta_{00}(\alpha_1^{00} + \alpha_2^{00} + \alpha_3^{00}) \\ f_1^3 = \beta_{10} - \beta_{00} \\ \\ f_1^0 = -\beta_{11}\alpha_1^{11}\alpha_2^{11}\alpha_3^{11} + \beta_{01}\alpha_1^{01}\alpha_2^{01}\alpha_3^{01} \\ f_1^1 = \beta_{11}(\alpha_1^{11}\alpha_2^{11} + \alpha_1^{11}\alpha_3^{11} + \alpha_2^{11}\alpha_3^{11}) - \beta_{01}(\alpha_1^{01}\alpha_2^{01} + \alpha_1^{01}\alpha_3^{01} + \alpha_2^{01}\alpha_3^{01}) \\ f_1^2 = -\beta_{11}(\alpha_1^{11} + \alpha_2^{11} + \alpha_3^{11}) + \beta_{01}(\alpha_1^{01} + \alpha_2^{01} + \alpha_3^{01}) \\ f_1^3 = \beta_{11} - \beta_{01} \end{array}\right\} \tag{8}$$

Next, we define the $h_{ij}(x)$ polynomials as

$$h_{ij}(x) = \beta_{ij}\alpha^{(ic+j)(n-d)}x^{n-d} - \beta_{ij} \tag{9}$$

with $i = \{0,1\}$, $j = \{0,1\}$ the integer value $c = 2$ and $\alpha = 2$ a primitive element of $\mathbb{F}_{13}$, so

$$h_{ij}(x) = \beta_{ij}2^{(i2+j)(n-d)}x^{n-d} - \beta_{ij} \tag{10}$$

Now by plugging (10) in (7), the equation system (8) becomes:

$$(i=1, j=0) \qquad \left.\begin{array}{l} f_1^0 = -\beta_{10} + \beta_{00} \\ f_1^3 = 2^6\beta_{10} - \beta_{00} \end{array}\right\} \tag{11}$$

$$(i=1, j=1) \qquad \left.\begin{array}{l} f_1^0 = -\beta_{11} + \beta_{01} \\ f_1^3 = 2^9\beta_{11} - 2^3\beta_{01} \end{array}\right\} \tag{12}$$

We take for instance (11) (taking (12) leads to the same result). As seen in (4), we have that:

$$f_3^0 = 0$$

now taking $\beta_{00} = 1$ and using (5), yields

$$(i=1, j=0) \qquad \left.\begin{array}{l} f_1^3 = 0 \\ \beta_{00} = 1 \\ \beta_{10} = 12 \\ f_1^0 = -12 + 1 = 2 \end{array}\right\} \tag{13}$$

Therefore,

$$\begin{array}{l} f_0(x) = 0 \\ f_1(x) = 2 \end{array} \tag{14}$$

Using these values in (12):

$$(i=1, j=1) \qquad \left.\begin{array}{l} 2 = -\beta_{11} + \beta_{01} \\ 0 = 2^9\beta_{11} - 2^3\beta_{01} \end{array}\right\} \tag{15}$$

solving, we have that

$$\beta_{01} = 1 \quad \text{and} \quad \beta_{11} = 12 \tag{16}$$

Which yields

$$\begin{array}{ll} h_{00}(x) = x^3 - 1 & = x^3 + 12 \\ h_{01}(x) = 2^3 x^3 - 1 & = 8x^3 + 12 \\ h_{10}(x) = 12 \cdot 2^6 x^3 - 12 & = x^3 + 1 \\ h_{11}(x) = 12 \cdot 2^9 x^3 - 12 & = 8x^3 + 1 \end{array} \tag{17}$$

Finally, using (7) we have

$$\begin{array}{l} g_0(x) = 12x^3 + 1 \\ g_1(x) = 5x^3 + 1 \end{array} \tag{18}$$

We have arrived at Coalition 1: $f_0(x) = 0, f_1(x) = 2$ and Coalition 2: $g_0(x) = 12x^3 + 1, g_1(x) = 5x^3 + 1$.

Encoding these polynomials, we have that for Coalition 1:

$$\begin{array}{l} \mathbf{f_0} = (0,0,0,0,0,0,0,0,0,0,0,0) \\ \mathbf{f_1} = (2,2,2,2,2,2,2,2,2,2,2,2) \end{array}$$

and for Coalition 2:

$$\begin{array}{l} \mathbf{g_0} = (0,6,0,2,6,6,9,9,0,2,9,2) \\ \mathbf{g_1} = (6,2,6,9,2,2,0,0,6,9,0,9) \end{array}$$

It is clear that both coalitions can create the same descendant (false fingerprint):

$$(0,2,0,2,2,2,0,0,0,2,0,2)$$

6 Conclusions

In this paper we have discussed the IPP in Reed-Solomon codes. The goal of our work was to answer a question by Silverberg et al. in [10, 11] inquiring whether all c-IPP Reed-Solomon codes are also c-TA codes. By expressing the IPP algebraically through the definition of a suitable set of polynomials, we have shown that for a large family of Reed-Solomon codes this is in fact true. That is, all $[n,k,d]$ Reed-Solomon codes defined over a field that contains the $n-d$ roots of unity are IPP codes if and only if they are also TA codes.

It is surprising that from our results it seems that the IPP characteristics of a Reed-Solomon code lie solely in the field over which the code is defined. To devise the exact extension of this dependence will be a subject of further research.

References

1. Alon, N., Fischer, E., Szegedy, M.: Parent-identifying codes. J. Comb. Theory, Ser. A **95**(2), 349–359 (2001)
2. Alon, N., Stav, U.: New bounds on parent-identifying codes: The case of multiple parents. Combinatorics, Probability & Computing **37**(6), 795–807 (2004)
3. Barg, A., Cohen, G., Encheva, S., Kabatiansky, G., Zémor, G.: A hypergraph approach to the identifying parent property: the case of multiple parents. Tech. rep., DIMACS 2000-20 (2000)
4. Barg, A., Kabatiansky, G.A.: A class of i.p.p. codes with efficient identification. J. Complexity **20**(2-3), 137–147 (2004)
5. Chor, B., Fiat, A., Naor, M.: Tracing traitors. Advances in Cryptology-Crypto'94, LNCS **839**, 480–491 (1994)
6. Chor, B., Fiat, A., Naor, M., Pinkas, B.: Tracing traitors. IEEE Trans. Inform. Theory **46**, 893–910 (2000)
7. Fernandez, M., Soriano, M.: Algorithm to decode identifiable parent property codes. Electronics Letters **38**(12), 552–553 (2002)
8. Hollmann, H.D.L., van Lint, J.H., Linnartz, J.P., Tolhuizen, L.M.G.M.: On codes with the Identifiable Parent Property. J. Combinatorial Theory **82**(2), 121–133 (1998)
9. Sarkar, P., Stinson, D.R.: Frameproof and IPP codes. Lecture Notes in Computer Science **2247**, 117–127 (2001). URL citeseer.ist.psu.edu/486714.html
10. Silverberg, A., Staddon, J., Walker, J.: Efficient traitor tracing algorithms using list decoding. ASIACRYPT 2001, LNCS **2248**, 175 ff. (2001)
11. Silverberg, A., Staddon, J., Walker, J.L.: Applications of list decoding to tracing traitors. IEEE Transactions on Information Theory **49**(5), 1312–1318 (2003)
12. Staddon, J.N., Stinson, D.R., Wei, R.: Combinatorial properties of frameproof and traceability codes. IEEE Trans. Inform. Theory **47**(3), 1042–1049 (2001)
13. Tô, V.D., Safavi-Naini, R.: On the maximal codes of length 3 with the 2-identifiable parent property. SIAM J. Discrete Math. **17**(4), 548–570 (2004)
14. van Trung, T., Martirosyan, S.: New constructions for ipp codes. Des. Codes Cryptography **35**(2), 227–239 (2005)