A Traceable Block Cipher

Olivier Billet and Henri Gilbert

France Télécom R&D
38-40, rue du Général Leclerc
92794 Issy les Moulineaux Cedex 9 - France
{olivier.billet,henri.gilbert}@francetelecom.com

Abstract. In this paper[1] we propose a new symmetric block cipher with the following paradoxical traceability properties: it is computationally easy to derive many equivalent secret keys providing distinct descriptions of the same instance of the block cipher. But it is computationally difficult, given one or even up to k equivalent keys, to recover the so called meta-key from which they were derived, or to find any additional equivalent key, or more generally to forge any new untraceable description of the same instance of the block cipher. Therefore, if each legitimate user of a digital content distribution system based on encrypted information broadcast (e.g. scrambled pay TV, distribution over the Internet of multimedia content, etc.) is provided with one of the equivalent keys, he can use this personal key to decrypt the content. But it is conjectured infeasible for coalitions of up to k traitors to mix their legitimate personal keys into untraceable keys they might redistribute anonymously to pirate decoders. Thus, the proposed block cipher inherently provides an efficient traitor tracing scheme [4]. The new algorithm can be described as an iterative block cipher belonging to the class of multivariate schemes. It has advantages in terms of performance over existing traitor tracing schemes and furthermore, it allows to restrict overheads to one single block (*i.e.* typically 80 to 160 bits) per encrypted content payload. Its strength relies upon the difficulty of the "Isomorphism of Polynomials" problem [17], which has been extensively investigated over the past years. An initial security analysis is supplied.

Keywords: traitor tracing, block ciphers, Matsumoto-Imai, multivariate cryptology, symmetric cryptology, collusion resistance.

1 Introduction

One of the most employed digital content distribution methods consists in broadcasting encrypted information. Applications include pay TV systems, server-based services for the distribution of pre-encrypted music, videos, documents or programs over the Internet, distribution of digital media such as CDs or DVDs, and more generally, conditional access systems. In content distribution systems broadcasting encrypted information, each user is equipped with a "decryption box" which may be a smart card combined with an unscrambling device as in

[1] This paper was submited to the Asiacrypt 2003 conference.

several existing pay TV systems, or even of software on a personal computer. The decryption box of each legitimate user is provided with a decryption key, allowing him to recover the plaintext content from the broadcast information during some validity period or for a given subset of the content. The delivery and update of decryption keys may be performed using various key distribution methods and is generally subject to the payment of subscriptions, digital right management licenses, etc.

The following security problem arises in this setting: if any legitimate user manages to recover the decryption key contained in his decryption box or to duplicate the keyed decryption software, then he can redistribute it to illegitimate users, allowing them to get the plain content as the legitimate users, without having to pay any subscription, digital right management license, etc. This quite often represents a much more serious threat than the redistribution of the plaintext content, which is so far not considered very practical in contexts like pay-TV. The use of tamper resistant devices (e.g. smart cards) to store decryption keys and associated algorithm(s) obviously helps protecting these systems, but can hardly be considered a sufficient countermeasure to entirely prevent this kind of attacks. Over the past years, more and more sophisticated attacks against tamper resistant devices have emerged—e.g. side-channel attacks, see for instance [12]. Because attacking a single decryption box may lead to massive fraud, attackers can afford using sophisticated and expensive attacks, so that countermeasures proposed in other contexts will often be ineffective for encrypted content broadcast systems.

Traitor tracing provides a natural countermeasure to prevent the decryption key redistribution threat described above. The concept of traitor tracing scheme was first introduced by B. Chor, A. Fiat and M. Naor in the seminal paper [4] and we use as far as possible the same terminology to describe the proposed scheme. In traitor tracing schemes, each legitimate user is provided with a unique personal decryption key which unambiguously identifies him, while enabling him to decrypt the broadcast information. The system must accommodate a large number N of users and it must be infeasible for any coalition of up to k legitimate users to mix their personal keys into a new untraceable description of the decryption key. Most of the traitor tracing schemes proposed so far, e.g. those described in [4], [14] and [18] are combinatorial in nature. Each legitimate user is provided with several base keys, which together form his personal key and the broadcast information contains large overheads of encrypted values under some of the base keys, allowing legitimate users to recover a content decryption key. A non-combinatorial alternative, namely a public key encryption scheme in which there is one public encryption key but many private decryption keys, was proposed by D. Boneh and M. Franklin in [3]. It has the advantage to avoid large overheads and to have very small decryption keys. However, the performance of this scheme is extremely sensitive to the maximum number k of tolerated colluding traitors, since the data expansion factor of the public key encryption is proportional to k.

The approach developed in this paper is non combinatorial in nature and has stronger connection with the one developed in [3] than with combinatorial schemes, up to the essential difference that we construct an untraceable symmetric cipher rather than an untraceable asymmetric cipher. The proposed cipher has the paradoxical property that many equivalent secret keys (used for decryption purposes) can be generated, while it is conjectured to be computationally impossible, given at most k equivalent secret keys, either to forge another untraceable equivalent secret key or to reconstruct the "meta key" from which the original equivalent secret keys were derived. More precisely, the knowledge of the meta key allows to efficiently determine at least one of the equivalent secret keys used to forge the new description.

The proposed construction can be described as an iterative block cipher. Its strength relies upon the intractability of the "Isomorphism of Polynomials," a problem which has been extensively investigated over the past years [2, 11, 17] and which conjectured intractability has not been directly affected by recent advances in the cryptanalysis of multivariate schemes like HFE [9, 10]. One of the advantages of the proposed scheme is to avoid generated overhead compared to the combinatorial approach taken in [3] where the data expansion is proportional to k. Another advantage is the intrinsic structure which is rather close to the one of usual block ciphers, so that the performance of the cipher in encryption/decryption modes is better than for existing traitors tracing schemes. Also the proposed scheme is much less sensitive to the maximum number k of traitors tolerated in a coalition, or to the maximum number of users N in the system. On the negative side, one should mention that the tracing procedures described in this paper require the knowledge of the description of the decryption function owned by a pirate. Thus no "black box" tracing procedure limiting interaction with the pirate decoder to "oracle queries" is provided. Another limitation of the proposed algorithm is that as usual in symmetric cryptography, no provable reduction to the difficulty of a well studied mathematical problem (e.g. the isomorphism of polynomial problem) could be found. Thus, the security analysis we supply can only achieve the next desirable goal, *i.e.* investigate various attack strategies and make sure that identified attacks are thwarted. Because of the higher requirements on a traceable cipher, risks are obviously much higher than for usual symmetric ciphers.

This paper is organized as follows. In Section 2, we describe the requirements on a symmetric cipher with an associated non-combinatorial traitor tracing scheme. In Section 3, we describe the proposed iterative block cipher construction and the associated traitor tracing scheme. Section 4 provides an initial security analysis. Section 5 addresses performance issues and provides an example instance of the proposed algorithm with explicit practical parameter values, in order to stimulate improved cryptanalysis. Section 6 concludes the paper.

2 Traceable Block Ciphers: Requirements and Operation

Let us denote by $F_{\mathcal{K}}$, $\mathcal{K} \in \mathbf{K}$ a symmetric block cipher of block size l, *i.e.* a key-dependent function from the set $\{0,1\}^l$ of l-bit input values to itself. As will be seen in the sequel, it is not required that $F_{\mathcal{K}}$ be easy to invert. It is not even an absolute requirement that the function $F_{\mathcal{K}}$ be one to one, although the block ciphers proposed in this paper are actually one to one and can be inverted: in practice they are operated in the forward direction alone, except in some traitor tracing procedures.

A traitor tracing scheme for N users associated with a traceable symmetric block cipher $F_{\mathcal{K}}$ consists of the following components:

- **A user initialization scheme** deriving users' secret keys $(\mathcal{K}_j)_{j=1,\dots,N}$ from a meta key $\mathcal{K} \in \mathbf{K}$. All user secret keys $\mathcal{K}_j$ must be distinct (though equivalent) descriptions $F_{\mathcal{K}_j}$ of the meta function $F_{\mathcal{K}}$. Each description $F_{\mathcal{K}_j}$ must allow to efficiently compute $F_{\mathcal{K}}$ in the forward direction.
- **Encryption and decryption processes**, respectively used by the operator of the broadcast distribution system to encrypt some digital content using $F_{\mathcal{K}}$, and by the legitimate user j to decrypt this content using his recovery key $\mathcal{K}_j$ through the associated description $F_{\mathcal{K}_j}$ of $F_{\mathcal{K}}$. As explained in [4], the structure of the broadcast information typically consists of pairs $(\mathrm{EB}_i, \mathrm{CB}_i)$ of an overhead information named "enabling block" and an encrypted content block named "cipher block." The enabling block is used to generate a symmetric key, hereafter called "control word," to decrypt the cipher block via an additional symmetric scheme S, like for instance AES or one-time pad. As said before, $F_{\mathcal{K}}$ needs not to be invertible: it is used in the forward direction in both the encryption and decryption processes.
- **A tracing procedure** allowing the owner of the meta key, when provided with any pirate description of the decryption function forged by any coalition of up to k traitors, to trace at least one traitor of the coalition.

In this setting, the meta key's holder creates cipher blocks CB_i from blocks of plain text content B_i using an additional symmetric scheme S and enabling blocks EB_i (produced for instance by a pseudo-random generator) via the formula $\mathrm{CB}_i := S_{\mathrm{CW}_i}(\mathrm{B}_i)$, where the control words CW_i are derived from the enabling blocks using the traceable block cipher $\mathrm{CW}_i := F_{\mathcal{K}}(\mathrm{EB}_i)$.

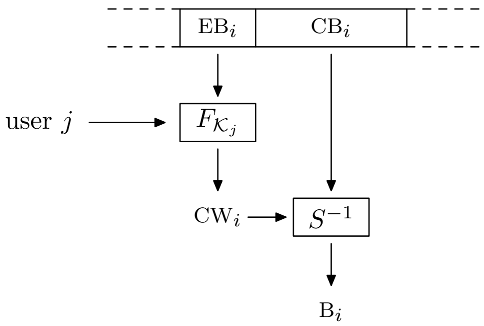


Fig. 1. Scheme's Architecture

The operations performed by legitimate users to decrypt these content blocks are summarized in Fig. 1. User j first derives the control word CW_i from the enabling block EB_i via his description $F_{\mathcal{K}_j}$ of the meta function: $\mathrm{CW}_i = F_{\mathcal{K}_j}(\mathrm{EB}_i)$. Then he uses the control word CW_i to decrypt the cipher block CB_i via the additional symmetric scheme S and recovers associated block(s) of plain content $\mathrm{B}_i = S^{-1}_{\mathrm{CW}_i}(\mathrm{CB}_i)$. For instance, $\mathrm{B}_i = \mathrm{CB}_i \oplus \mathrm{CW}_i$ when S is the one time pad algorithm, and $\mathrm{B}_i = \mathrm{AES}^{-1}_{\mathrm{CW}_i}(\mathrm{CB}_i)$ when S is the AES. In this context, control words must be frequently generated to prevent attacks by redistribution of these control words to pirate decryption boxes from being much easier than the redistribution of plaintext. This is difficult to achieve with existing combinatorial traitor tracing schemes due to the large data expansion incurred by such schemes. Another consequence is that the throughput (bit/s) of the $F_{\mathcal{K}}$ block cipher must be as close as possible to the throughput of classical block ciphers such as AES and much larger than the one of asymmetric ciphers such as RSA. An additional requirement for systems where $\mathcal{K}$ needs to be updated frequently, e.g. to manage dynamic modifications of lists of subscribers, is that each description $F_{\mathcal{K}_j}$ be reasonably short for the distribution via any symmetric encryption or key distribution algorithm to be practical.

In order for the content distribution system to resist attacks against the decryption scheme, the descriptions $F_{\mathcal{K}_j}$ must satisfy the usual security requirements of a block cipher. This implies that given any set of $F_{\mathcal{K}_j}$ input/output pairs with known, chosen or even adaptively chosen input values an adversary could obtain, it must be computationally infeasible for this adversary to predict any additional $F_{\mathcal{K}_j}$ input/output pair with a non negligible success probability. In particular input/outputs pairs must not reveal $\mathcal{K}_j$ or any other equivalent description of $F_{\mathcal{K}}$.

The last and most demanding requirement is the existence of an efficient traitor tracing procedure for the owner of the meta key $\mathcal{K}$. Our definition of a traitor tracing scheme follows the one proposed in the seminal paper [4]. We do not require the traitor tracing scheme to be black box (*i.e.* to be operable using say only inputs EB_i and outputs CW_i of the key distribution function). We restrict ourselves to traitor tracing scenarios where an authority is able to access the description of the description of $F_{\mathcal{K}}$ contained in the pirate decryption box. Note that it does not seem unrealistic to assume that decryption boxes of pirate users can be tampered by an authority, taking into account the fact that traitor tracing is only needed if the decryption boxes of legitimate users can be tampered. Traitor tracing requirements can be informally stated as follows. Attacks by any coalition of up to k traitors should be traceable, that is k traitors able to access their individual descriptions $F_{\mathcal{K}_j}$ should not be computationally able to forge any additional description F' from their k equivalent descriptions $F_{\mathcal{K}_j}$ without revealing at least one of their $\mathcal{K}_j$—and thus the identity j of one of the traitors. We further require that the probability for the tracing procedure applied to any k-traitors coalition to either output no suspected traitor (non detection) or to output the identity j of an innocent user (false alarms) be negligible.

3 Description of the Traceable Scheme

Among the requirements identified in the former Section, the most demanding one is not the existence of many equivalent descriptions of the symmetric function $F_{\mathcal{K}}$—this is frequent in symmetric cryptography, see for instance [1]—but the property that the provision to a user of one of these numerous representations $F_{\mathcal{K}_j}$ should not disclose information allowing him to construct any other representation of $F_{\mathcal{K}}$ unrelated to $\mathcal{K}_j$. In other words, the meta key $\mathcal{K}$ must act as a kind of trapdoor allowing to perform other operations than those allowed by the descriptions $F_{\mathcal{K}_j}$ of $F_{\mathcal{K}}$. Thus, even in the symmetric setting considered in this paper, public key cryptography properties are required and generic block ciphers will not be usable like in the case of combinatorial traitor tracing schemes. However we would like to keep performance advantages of symmetric cryptography since generation of control words at high rate is necessary for the security of the system.

Multivariate cryptography appears to be a natural candidate to meet these requirements. As a matter of fact, features of this recently developed family of algorithms are to many extents intermediate between those of public key algorithms (e.g. trapdoors) and those of secret key algorithms. Many of them can be described as iterative ciphers resulting of the composition of several rounds, and their complexity is substantially lower than the one of usual public key ciphers and not much higher than the one of usual block ciphers. Typical examples of multivariate algorithms are C^* proposed by T. Matsumoto and H. Imai in [13], SFLASHv2 (one of the Nessie finalists [19]), and HFE [16]. All the schemes mentioned above rely on the intractability of the so-called "Isomorphism of Polynomials" problem for the *secret key recovery*. See [7] for more information about known attacks against this problem. The C^* scheme was attacked by Patarin in [15] and Dobbertin independently, but these attacks do not allow to recover the secret key and thus to break the underlying IP problem. An attack allowing to solve the IP problem underlying some instances of HFE, using so-called relinearization techniques was published by Kipnis and Shamir in 1999 [11], and appears to be also applicable to the IP problem underlying some instances of the basic (quadratic) version of C^*. More recently, enhanced decryption or signature forgery attacks against HFE and more generally various multivariate cryptosystems have been proposed [8, 6, 9, 10]. But none of these recent attacks allows to recover the secret key and to break the underlying IP problem. Thus in summary, as far as we know, the best known attacks against the IP problem underlying multivariate schemes are those described in [7, 11].

3.1 Building Blocks

Let us briefly recall the basic quadratic C^* from which the building block of our scheme is directly derived by generalizing it to monomials of higher degree. It involves the following elements:

- A finite field $\mathbb{K} = \mathbb{F}_q$ of size q.

- An extension $\mathbb{L}$ over $\mathbb{K}$ of degree n, with a defining primitive polynomial $P(X)$ of degree n such that $\mathbb{L} = \mathbb{K}[X]/(P(X))$. We will represent elements of $\mathbb{L}$ as n-tuples $(a_0, \ldots, a_n)$ of $\mathbb{K}$ through the usual identification function $\varphi : (a_0, \ldots, a_n) \mapsto \sum_{i=0}^{n} a_i X^i \pmod{P(X)}$.
- A private key made of two linear one to one mappings s and t from $\mathbb{K}^n$ to itself and an integer θ such that $q^\theta + 1$ be prime to $q^n - 1$.
- A public key $G = t \circ \varphi^{-1} \circ E_\theta \circ \varphi \circ s$, published as a system of n multivariate polynomials in n variables, where E_θ is a monomial function defined to be $\mathbb{L} \to \mathbb{L}$, $a \mapsto a^{1+q^\theta}$. Assuming the trapdoor (s, t) unknown, function G was believed to be one-way, but J. Patarin showed in [15] that it can be computationally inverted. However, one-wayness is not needed in our scheme.

The actual building blocks of our construction are higher degree variants of C^* obtained by considering a more generic—but still monomial—function E, namely $E_\Theta : \mathbb{L} \to \mathbb{L}$, $a \mapsto a^{1+q^{\theta_1}+\ldots+q^{\theta_{d-1}}}$ where d is a fixed integer and Θ is a $(d-1)$-tuple $(\theta_1, \ldots, \theta_{d-1})$ such that $q^n - 1$ be prime to $1 + q^{\theta_1} + \ldots + q^{\theta_{d-1}}$, hereafter called the degree of the building block G. Indeed, G can be described as a system of n multivariate polynomial equations as suggested in Fig. 2, and the polynomials P_i involved have total degree d. For instance, in the special case where $d = 3$, G can be described as $(i = 1, \ldots, n)$:

$$y_i = \sum_{0 \le j,k,l \le n-1} \alpha_{i,j,k,l} x_j x_k x_l + \sum_{0 \le j,k \le n-1} \beta_{i,j,k} x_j x_k + \sum_{0 \le j \le n-1} \gamma_{i,j} x_j \quad . \tag{1}$$

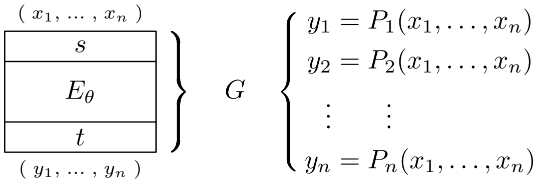

$(x_1, \cdots, x_n)$
s
E_θ
t
$(y_1, \cdots, y_n)$
G
$y_1 = P_1(x_1, \ldots, x_n)$
$y_2 = P_2(x_1, \ldots, x_n)$
$\vdots \quad \vdots$
$y_n = P_n(x_1, \ldots, x_n)$

Fig. 2. An extended C^* building block.

The basic idea underlying the proposed traitor tracing scheme is to use several of those extended C^* instances as building blocks for our construction and to take opportunity of the commutativity of the various monomial functions E_θ involved—that is $E_{\theta_1} \circ E_{\theta_2} = E_{\theta_2} \circ E_{\theta_1}$ for all θ_1, θ_2.

3.2 Meta Key, Users' Keys

Let us keep the notation of the previous Section. Moreover, let r be the number of building blocks. The meta secret key $\mathcal{K}$ is defined as the set of two one to one linear mappings s and t from $\mathbb{K}^n$ to itself, and a collection of r $(d-1)$-tuples Θ_i such that all the values $1 + q^{\theta_{1,i}} + \ldots + q^{\theta_{d-1,i}}$ for $i = 1, \ldots, r$ be distinct. Then the function $F_\mathcal{K}$ is defined as $F_\mathcal{K} = s \circ E_{\Theta_r} \circ \cdots \circ E_{\Theta_2} \circ E_{\Theta_1} \circ t$.

Now assign to each user j a private key $\mathcal{K}_j$ generated after the meta key $\mathcal{K}$ using a set of $r-1$ linear one to one mappings $L_{1,j}$, $\ldots$, $L_{r-1,j}$ from $\mathbb{K}^n$ to itself, and a permutation σ_j of the set $\{1,\ldots,n\}$. The user gets his key $\mathcal{K}_j$ as a list of functions $G_{1,j}$, $\ldots$, $G_{r,j}$, which are provided as systems of n multivariate equations of homogeneous degree as described in Figs. 2 and 3.

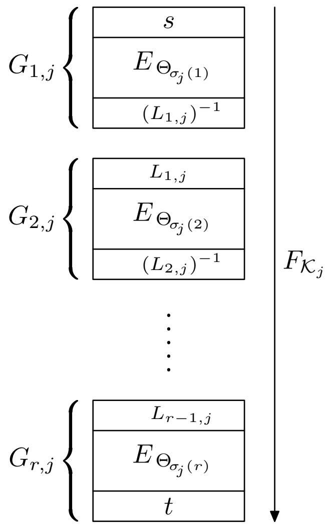


Fig. 3. Description $F_{\mathcal{K}_j} = G_{r,j} \circ \cdots \circ G_{2,j} \circ G_{1,j}$

A user initialization scheme needed to derive a user's key from the meta key $\mathcal{K}$ follows. From any input j one creates the permutation σ_j and the $r-1$ one to one mappings $L_{i,j}$ by any pseudo- random generation mechanism or by any diversification algorithm.

We can now check that the users' functions $F_{\mathcal{K}_j}$ are distinct but equivalent descriptions of the meta function $F_{\mathcal{K}}$. Indeed, for each user j, the one to one mappings at the end of $G_{k,j}$ and at the beginning of $G_{k+1,j}$ cancel out, and since the functions E_Θ are commuting, the effect of the permutation is annihilated.

3.3 Encryption and Decryption

In order to encrypt a digital content, the station may broadcast enabling block and cipher block pairs $(\mathrm{EB}_i, \mathrm{CB}_i)$ produced with the help of any additional symmetric algorithm S_{CW} where the symmetric key is the control word generated as $\mathrm{CW} := F_{\mathcal{K}}(\mathrm{EB}_i)$. Thus, the construction is given by $\mathrm{CB}_i := S_{F_{\mathcal{K}}(\mathrm{EB}_i)}(\mathrm{B}_i)$, where B_i denotes the content block. Now any user j can recover the content block by following a similar procedure, that is by computing $\mathrm{B}_i := S^{-1}_{F_{\mathcal{K}_j}(\mathrm{EB}_i)}(\mathrm{CB}_i)$.

3.4 Traitor Tracing Procedure

The procedure to identify traitors relies upon the two following claims which are substantiated in the security analysis given in the next section.

Claim 1. When the leakage originates from a single traitor l, the analysis of the description F' constructed by the traitor based on his description $F_{\mathcal{K}_l}$ allows the authority to decompose F' in r components G'_1 to G'_r such that $F' = G'_r \circ \cdots \circ G'_1$. Moreover, each G'_i can be split as the composition of the functions G_i of the traitor and other "parasitic" functions which may differ from the identity function. Thus, the analysis reveals the order of composition of the functions G_i which in turn reveals the identity of the traitor through the knowledge of σ_l.

This first claim allows an authority provided with the meta key $\mathcal{K}$ to efficiently derive the permutation σ_l associated to the description $F_{\mathcal{K}_l}$ of the traitor from the leaked function F', and thus to recover the identity l of the traitor.

Claim 2. When the leakage originates from a coalition of at most k traitors, the analysis of the description F' constructed by the k colluding traitors allows to decompose F' in r components G'_1 to G'_r such that the middle $r - 2\rho$ values come from "parasitized" functions G_i of a single traitor, for a well chosen ρ.

This second claim allows, by properly choosing the parameters of the system, specially ρ which exact definition is to be given in the next Section, to recover the identity of one of the traitors—say j—by deriving the values of the permutation σ_j on the set of integers $[\rho, r - \rho]$ from the values of the functions $G_{\rho,j}$ to $G_{r-\rho,j}$ alone. To achieve this goal, we must ensure that the middle part of the pirate description F' originates from the middle parts ρ to $r - \rho$ of one single traitor, while mixing traitors' descriptions in the ranges $[1, \rho]$ and $[r - \rho, r]$ can still be tolerated.

4 Security Discussion

4.1 The IP Problem

The security of the proposed traceable iterated symmetric cipher relies to a large extent upon the security of special instances of the "Isomorphism of Polynomials" problem—hereafter called IP—namely the problem of *finding the hidden monomial* of the extended Matsumoto-Imai C^* scheme described in Section 3.1.

The IP problem with two secrets—see also [7, 17]—consists in finding a pair (s, t) of one to one linear mappings between two sets A and B of multivariate polynomial equations of total degree d over a finite field $\mathbb{K}$. Denoting by $x = (x_1, \ldots, x_n)$ an element of $\mathbb{K}^n$, we can write $y = A(x)$ as a system of polynomial equations:

$$\begin{cases} y_1 &= P_1(x_1, \ldots, x_n) \\ y_2 &= P_2(x_1, \ldots, x_n) \\ \vdots & \;\;\vdots \\ y_n &= P_n(x_1, \ldots, x_n) \;, \end{cases}$$

and similarly for B. In this setting the IP problem consists in finding a pair of one to one linear mappings s and t such that:

$$B\bigl(s(x)\bigr) = t\bigl(A(x)\bigr). \tag{2}$$

This problem is assumed to be difficult and it has been shown to be at least as hard as the "Graph Isomorphism" problem. Even for very special instances complexity remains high [2, 6]. Note also that an efficient solution to the IP problem would lead to an efficient attack on SFLASHv2 [19] that has been selected by the European Nessie project.

4.2 Resisting Attacks Against the Decryption Scheme

As explained in Section 2, the descriptions $F_{\mathcal{K}_j}$ of any user j must satisfy the usual security requirements of block ciphers. In particular, given any realistic number of input/output pairs of $F_{\mathcal{K}_j}$ corresponding to chosen or adaptively chosen input values, it must be computationally infeasible to infer any additional output value. Based on an investigation of the most natural attack strategies, we conjecture that this property is satisfied provided that:

1. Parameters q and n be chosen so that even if the monomial functions E_{Θ_1}, E_{Θ_2}, ..., E_{Θ_n} can be guessed, solving the IP problem which consists of guessing s and t given a sufficient large number of input/output pairs of $F_{\mathcal{K}_j}$ be intractable. Based on the results in [7, 2] we expect this condition to be satisfied provided that the complexity q^n of the best know attack be large enough, say at least 2^{80}. Since an enhanced attack of complexity $q^{n/2}$ is reported in the quadratic case in [7], an even more conservative choice would be to consider $q^n > 2^{160}$ in order to prevent a generalization of this attack to other instances of IP.
2. The value q^D, where D is the degree of the system of polynomial equations in n variables representing any $F_{\mathcal{K}_j}$ be large enough, say at least 2^{80}, to prevent attacks based on higher order derivation. Indeed, this would allow an attacker to predict one more output given an affine set of q^{D+1} input values and and all but one of their corresponding outputs. D is about nq when r is large enough and q is the size of the finite field $\mathbb{K}$;
3. The number of monomials of the system of n polynomial equations in n variables representing any $F_{\mathcal{K}_j}$, which is usually close to $n\binom{n+D-1}{D}$, be large enough to prevent an attacker from recovering the coefficients of this system using linear algebra and a sufficient number of input/output pairs of $F_{\mathcal{K}_j}$.

4.3 Tracing Single Traitor's Pirate Description

We anticipate that in trying to produce an untraceable version of his description $F_{\mathcal{K}_j}$, a traitor j would adopt one of the following strategies:

1. Try to find one of the $r + 1$ one to one linear mappings s, $L_{1,j}$, $L_{2,j}$, ..., $L_{r-1,j}$ and t, hidden to the attacker j. If an attacker j could recover one

of these $r+1$ linear mappings, say $L_{l,j}$, this would obviously allow him to incrementally recover all the $L_{i,j}$ for $i < l$, and all the $L_{i,j}$ for $i > l$, using the information provided by the mappings $G_{1,j}$ to $G_{r,j}$, and thus to recover the value of $\mathcal{K}_j$ and to easily produce variants of his description $F_{\mathcal{K}_j}$ in an untraceable manner. Conversely, we conjecture this to be as hard as solving the IP problem of at least one of the $G_{i,j}$. The complexity of the best attacks reported in [7] are $O(q^n)$ in case $d > 2$ and $O(q^{n/2})$ in case $d = 2$.

2. Try to directly use the functions $G_{\cdot,j}$ without analyzing them, by modifying them so as to produce a concealed variant of the original description by composing the basic blocks $G_{\cdot,j}$ in the same order, but with "parasitic" functions whose effects eventually cancel out. That is the traitor tries to produce a sequence $(G'_{i,j})_{i\in[1,w]}$ with *two types* of blocks G': those which can be written as $\varphi_i \circ G_{i,j} \circ \psi_{i+1}$ and those that do not rely on the available $G_{i,j}$ blocks and are denoted by Π_i. These data must be such that the effects of adding/composing the φ, Π and ψ mappings to the original blocks $G_{i,j}$ eventually cancel out, that is so that $F_{\mathcal{K}_j} = G'_{w,j} \circ \cdots \circ G'_{1,j}$. (Please note that w can be greater than r because of the *second type* of blocks.) Also note that φ_i, ψ_i and Π_i have to be simple enough—for instance a reasonable number of monomials and a limited total degree—so that they could be easily constructed and efficiently computed.
3. Try to compose several blocks $G_{i,j}$ of his description. This attack is impossible as soon as the number of monomial in such composition is impractical. Since composition must be formally computed, $\binom{n+d-1}{d}$ terms must be formally put to the power of d which is quickly intractable. As will be seen in the sequel, composition of a small number of blocks $G_{i,j}$, say 2 of them, do not substantially complexify the tracing procedure. Therefore, only the composition of more than 3 blocks must be prevented.
4. Use a combination of any of the above strategies.

To trace traitor j from a pirate description $G'_1, \ldots, G'_w$, the authority proceeds as follows. First, note that G'_1 is necessarily of the form $\psi_1 \circ L_{1,j} \circ E_{\Theta_{\sigma_j(1)}} \circ s$, that is of the *first type*. The authority thus searches for $\sigma_j(1)$ by using its knowledge of s^{-1}, and all the $E^{-1}_{\Theta_i}$: it computes $G'_1 \circ s^{-1} \circ E^{-1}_{\Theta_i}$ for each i, and guesses the right value i by testing the "simplicity" of the resulting function by means of chosen input/output pairs. The simplicity is evaluated by estimating the degree and the number of monomials. In case of a correct guess, the function has a low degree and a predetermined number of monomials whereas in case of a bad guess the function has terms of high degree. Having guessed the value $\sigma_j(1)$, we denote it by $\alpha(1)$.

The authority then has to get rid of terms of *second type* Π_i, until another term of *first type* is found. This is done again by evaluating the simplicity of the successive compositions:

$$G'_2 \circ G'_1 \circ s^{-1} \circ E^{-1}_{\Theta_{\alpha(1)}} \circ E^{-1}_{\Theta_i} \;,$$
$$G'_3 \circ G'_2 \circ G'_1 \circ s^{-1} \circ E^{-1}_{\Theta_{\alpha(1)}} \circ E^{-1}_{\Theta_i} \;,$$
$$\vdots$$

each time for all i until a simple composed function is found. The authority then finds the value $\sigma_j(2)$ and denotes it by $\alpha(2)$. The process goes on iteratively and eventually gives the permutation σ_j allowing the authority to trace traitor j.

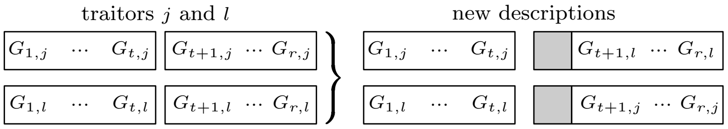
While choosing the parameters of the system, we will make it hard for an attacker to formally compose two extended C^* blocks and totally intractable to compose three of them. The composition of two consecutive blocks can be easily thwarted since the above guessing procedure remains valid when replacing $E_{\Theta_i}^{-1}$ by $E_{\Theta_i}^{-1} \circ E_{\Theta_j}^{-1}$ varying both i and j at the same time, thus allowing to trace such compositions of two blocks as well.

4.4 Tracing *k* Traitors' Pirate Descriptions

The best collusion strategy we identified for a coalition of at most k traitors provided with distinct descriptions $F_{\mathcal{K}_j} = G_{r,j} \circ \cdots \circ G_{1,j}$ associated with the same meta description $F_{\mathcal{K}}$ is the following one.

The basic idea is that the traitors may take advantage of the fact that the initial mapping s and the final mapping t are identical for every user. This could allow them to detect a partial collision between their respective hidden permutation σ. Let us take the example of two traitors j and l searching for such a collision. They know their first blocks begin with the same mapping s, and if their first functions $E_{\sigma_j(1)}$ and $E_{\sigma_l(1)}$ were equal, then blocks $G_{1,j}$ and $G_{1,l}$ would be equal up to a one to one linear mapping, namely $L_{1,j}^{-1} \circ L_{1,l}$. Otherwise it would not be a one to one linear mapping. This is easy to test and provides a way for a pair of traitors to guess if their permutations take the same values on 1, *i.e.* if $\sigma_j(1) = \sigma_l(1)$.

Now, whether they succeed or not in the last step, the pair of traitors go further in the process by checking whether $G_{2,j} \circ G_{1,j}$ and $G_{2,l} \circ G_{1,l}$ are equal up to another hidden one to one linear mapping. (Remember that the commutativity of the functions E_{Θ_i} makes this possible.) In case of success, this would allow them to deduce that the images of the unordered set $\{1, 2\}$ under both permutations are equal: $\sigma_j\big(\{1, 2\}\big) = \sigma_l\big(\{1, 2\}\big)$, and provide them with the value of $L_{2,j}^{-1} \circ L_{2,l}$. By iterating the process, the pair of traitors may identify any collision of their respective permutations on the set of integers $[1, t]$ for any $t \in [1, r]$, hereafter called a t-collision. Moreover, any detected t-collision provides a way to forge two new pirate descriptions by exchanging the first t components of their respective descriptions of the meta function $F_{\mathcal{K}}$, as shown in Fig. 4.



$$\Big(G_{1,j}, G_{2,j}, \ldots, (L_{t,j}^{-1} \circ L_{t,l})^{-1} \circ G_{t,j}, G_{t+1,l}, \ldots, G_{r,l}\Big) \ ,$$
$$\Big(G_{1,l}, G_{2,l}, \ldots, (L_{t,l}^{-1} \circ L_{t,j})^{-1} \circ G_{t,l}, G_{t+1,j}, \ldots, G_{r,j}\Big) \ .$$

Fig. 4.

Note that the traitors can search for all collisions. That is, when no u-collision was found for $u < t$, it still remains possible for them to find a t-collision, *when such a collision exists.* Of course, this scenario can be replayed with other traitor pairs, or even with the newly forged descriptions, leading to a possible great amount of untraceable pirate keys.

To avoid this situation, one encodes the identity of any user i in the values taken by the permutation σ_i on the middle interval $[\rho, r-\rho]$ of the original one $[1, r]$, for some well chosen $\rho < r/2$ so that the probability of *any* t-collision for $t \in [\rho, r-\rho]$ is arbitrarily small.

Obviously, attacks involving a single traitor can also be used by coalition of traitors in addition to the specific techniques discussed in this Section, but those can be handled the same way.

4.5 Non-Detection and False Alarms.

Let us derive the requirements the attack scenario of the previous Section puts on parameter ρ. First, for any traitors' pair, the probability that a t-collision holds is $1/\binom{r}{t}$. Thus the probability that a t-collision for a coalition of up to k traitors occurs for $t \in [\rho, r-\rho]$ is at most

$$P_k = \frac{k(k-1)}{2} \sum_{t=\rho}^{r-\rho} \frac{1}{\binom{r}{t}} \quad .$$

At the same time, permutations of users must be distinguishable from their values in the interval $[\rho, r-\rho]$. This implies that the number of distinct identities available for the system will be at most $M = r!/(2\rho)!$.

Now if the scheme needs to handle at most N users where $N < M$, and assuming a coalition of up to k traitors, the probability of non-detection (the authority detects a collusion, but no matching identity is found) is given by $P_{k,\mathrm{ND}} = (1 - N/M)\, P_k$ while the probability of false alarm (a wrong identity is pointed out) is given by $P_{k,\mathrm{FA}} = N/M\, P_k$. This comes from the fact that there are $(M-N)$ permutations that do not correspond to any valid identity.

5 Practical Example

We provide realistic example parameters such that the scheme accommodates $N = 10^6$ users. The field of operation $\mathbb{K}$ is taken to be $\mathrm{GF}(2^{16})$ so that $m = 16$ and $q = 2^{16}$. Moreover, we chose $n = 5$ and the degree of the monomials in an extended C^* block to be $d = 4$. There is a total of 32 distinct $(d-1)$-tuples Θ such that $1 + q^{\theta_1} + \ldots + q^{\theta_{d-1}}$ is prime to $q^n - 1$.

Letting $r = 32$ and $\rho = 13$ makes the probability of false alarms smaller than $2\,10^{-10}$ for any coalition of up to $k = 10$ traitors, smaller than $2.2\,10^{-8}$ for $k = 100$ traitors and smaller than $2.3\,10^{-6}$ for $k = 1000$. Probability of non-detection is smaller than $1.2\,10^{-7}$ when $k = 10$, smaller than $1.5\,10^{-3}$ when $k = 1000$. Other security requirements are met since $q^n = 2^{80}$ and furthermore the number of monomials in a building block of $F_{\mathcal{K}}$ is 350, so that in any formal composition of three of them the number of monomials is already more than $4\,10^6$, and in any formal composition of four blocks it is about 10^9.

With this choice of parameters, the total size of any description equivalent to $F_{\mathcal{K}}$ is 21,8 KB. Speed of encryption is essentially determined by the number of multiplications in $F_{\mathcal{K}_j}$ to be performed and can roughly be estimated as follows: the 70 terms $x_1^{\nu_1} \cdots x_5^{\nu_5}$ of total degree four can be computed once for each block and then multiplied by the appropriate leading coefficients of the polynomials describing each output variable of a block. So one can compute the 70 homogeneous terms of degree 4 in 85 multiplications in $\mathbb{K}$ and eventually compute $y_1, \ldots, y_5$ in at most $5{\cdot}70$ multiplications in $\mathbb{K}$. Since there are 32 blocks, that makes a total of about 15000 multiplications to process any $F_{\mathcal{K}_j}$ on the 80 bit input. Additionally, the size of the overhead in this example is obviously 80 bits.

We propose another realistic set of parameters, hopefully more conservative, for applications where storage and speed of encryption are less critical concerns. The scheme handles up to $N = 10^6$ users. The field of operation is taken to be $\mathrm{GF}(2^9)$, while the number of variables is set to $n = 19$ and the degree of the monomials is set to $d = 3$. There is a total of 190 distinct $(d-1)$-tuples Θ such that $1 + q^{\theta_1} + \ldots + q^{\theta_{d-1}}$ is prime to $q^n - 1$. Choosing $r = 33$ and $\rho = 10$ makes the probability of false alarms smaller than $1.4\,10^{-19}$ for any coalition of up to $k = 10$ traitors, smaller than $1.52\,10^{-15}$ for any coalition of up to $k = 1000$ traitors, and the probability of non-detection smaller than $5\,10^{-7}$ for any coalition of up to $k = 10$ traitors, smaller than $5.4\,10^{-3}$ for any coalition of up to $k = 100$ traitors. Security requirements are met since $q^n = 2^{171}$ and the number of monomials in a building block of $F_{\mathcal{K}}$ is 25270, so that in any formal composition of three of them the number of monomials is already more than $90\,10^6$ and in any formal composition of three blocks it is already more than $3\,10^{13}$. In that case, the size of any equivalent decryption key is 916 KB. The 1330 monomials can be computed in 1520 multiplications in $\mathbb{K}$ so that a building block requires 26790 multiplications and it takes about 900000 multiplications to evaluate any description $F_{\mathcal{K}}$ on the 171 bits of the input. The overhead is obviously of 171 bits.

k	10	100	1000
$P_{k,\mathrm{FA}}$	$< 2\,10^{-10}$	$< 2.2\,10^{-8}$	$< 2.3\,10^{-6}$
$P_{k,\mathrm{ND}}$	$< 1.2\,10^{-7}$	$< 1.5\,10^{-5}$	$< 1.5\,10^{-3}$

$N = 10^6$, $r = 32$, $\rho = 13$, $n = 5$, $\mathbb{F} = \mathrm{GF}(2^8)$.

k	10	100	1000
$P_{k,\mathrm{FA}}$	$< 1.4\,10^{-19}$	$< 1.5\,10^{-17}$	$< 1.6\,10^{-15}$
$P_{k,\mathrm{ND}}$	$< 5\,10^{-7}$	$< 5.4\,10^{-5}$	$< 5.4\,10^{-3}$

$N = 10^6$, $r = 33$, $\rho = 10$, $n = 19$, $\mathbb{F} = \mathrm{GF}(2^9)$.

Fig. 5. Summary of parameters and corresponding probabilities

6 Conclusion

A novel iterative block cipher which can be operated in a traceable manner has been introduced. The attacks investigated in our initial security analysis are easy to prevent by properly selecting system parameters. Improvements in these attacks are of course not precluded, since no reduction proof of the security to a well identified mathematical problem was found apart from obvious connection to the "Isomorphism of Polynomial" problem. Risks are obviously higher than for usual symmetric ciphers. Natural questions also arise: What security does the "Isomorphism of Polynomials" problem provide for small values of the number n of variables like those suggested in Section 5? Also, other building blocks could be considered, e.g. variants with two or more branches in each extended C^* block. Studying the effects of releasing the constraint that the monomial functions be distinct may lead to some performance improvements. We also note that since each user possesses an equivalent description, he is able to broadcast data to every other user. Besides traitor tracing, another interesting application of the proposed construction is whitebox cryptography [5]. Indeed, advantage can be taken from the fact that one can easily construct a huge number of equivalent descriptions, while those descriptions can be made arbitrarily large.

In its current shape, the proposed traceable block cipher has the advantage of being very insensitive to the maximum number of traitors tolerated while accommodating a large number of users. Due to its intrinsic block cipher structure and due to the fact that it does not generate any data expansion overhead, its implementation can be made very efficient.

References

1. Elad Barkan and Eli Biham, *In how many ways can you write Rijndael?*, available from the e-print at `http://eprint.iacr.org/2002/157/`.
2. Alex Biryukov, Christophe De Canniere, An Braeken, and Bart Preneel, *A Toolbox for Cryptanalysis: Linear and Affine Equivalence Algorithms*, Advances in Cryptology – EUROCRYPT 2003 (Eli Biham, ed.), Lecture Notes in Computer Science, vol. 2656, Springer-Verlag, 2003, pp. 33–50.
3. Dan Boneh and Matthew Franklin, *An Efficient Public Key Traitor Tracing Scheme*, Advances in Cryptology – CRYPTO '99 (Michael Wiener, ed.), Lecture Notes in Computer Science, vol. 1666, Springer-Verlag, 1994, pp. 338–353.
4. Benny Chor, Amos Fiat, and Moni Naor, *Tracing Traitors*, Advances in Cryptology – CRYPTO '94 (Yvo G. Desmedt, ed.), Lecture Notes in Computer Science, vol. 839, Springer-Verlag, 1994, pp. 257–270.
5. Stanley Chow, Philip Eisen, Harold Johnson, and Paul C. Van Oorschot, *White-Box Cryptography and an AES Implementation*, Selected Areas in Cryptography – SAC 2002 (K. Nyberg and H. Heys, eds.), Lecture Notes in Computer Science, vol. 2595, Springer-Verlag, 2002, pp. 250–270.
6. Nicolas Courtois, Louis Goubin, Willi Meier, and Jean-Daniel Tacier, *Solving Underdefined Systems of Multivariate Quadratic Equations*, Public Key Cryptography – PKC 2002 (David Naccache and Pascal Paillier, eds.), Lecture Notes in Computer Science, vol. 2274, Springer-Verlag, 2002, pp. 211–227.

7. Nicolas Courtois, Louis Goubin, and Jacques Patarin, *C^{*-+} and HM: Variations around two schemes of T. Matsumoto and H. Imai*, Advances in Cryptology – ASIACRYPT '98 (Kazuo Ohta and Dingyi Pei, eds.), Lecture Notes in Computer Science, vol. 1514, Springer-Verlag, 1998, pp. 35–49.
8. Nicolas Courtois, Alexander Klimov, Jacques Patarin, and Adi Shamir, *Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations*, Advances in Cryptology – EUROCRYPT 2000 (Bart Preneel, ed.), Lecture Notes in Computer Science, vol. 1807, Springer-Verlag, 2000, pp. 392–407.
9. Nicolas T. Courtois, Magnus Daum, and Patrick Felke, *On the Security of HFE, HFEv- and Quartz*, Public Key Cryptography – PKC 2003 (Yvo G. Desmedt, ed.), Lecture Notes in Computer Science, vol. 2567, Springer-Verlag, 2003, pp. 337–350.
10. Jean-Charles Faugère and Antoine Joux, *Algebraic Cryptanalysis of Hidden Field Equation (HFE) Cryptosystems Using Gröbner Bases*, Advances in Cryptology – CRYPTO 2003 (Dan Boneh, ed.), Lecture Notes in Computer Science, vol. 2729, Springer-Verlag, 2003, pp. 44–60.
11. Aviad Kipnis and Adi Shamir, *Cryptanalysis of the HFE Public Key Cryptosystem by Relinearization*, Advances in Cryptology – CRYPTO '99 (Michael Wiener, ed.), Lecture Notes in Computer Science, vol. 1666, Springer-Verlag, 1999, pp. 19–30.
12. Paul Kocher, Joshua Jaffe, and Benjamin Jun, *Differential Power Analysis*, Advances in Cryptology – CRYPTO '99 (Michael Wiener, ed.), Lecture Notes in Computer Science, vol. 1666, Springer-Verlag, 1999, pp. 388–397.
13. Tsutomu Matsumoto and Hideki Imai, *Public Quadratic Polynomial-tuples for Efficient Signature Verification and Message Encryption*, Advances in Cryptology – EUROCRYPT '88 (Cristoph G. Günther, ed.), Lecture Notes in Computer Science, vol. 330, Springer-Verlag, 1988, pp. 419–453.
14. Moni Naor and Benny Pinkas, *Threshold Traitor Tracing*, Advances in Cryptology – CRYPTO '98 (Hugo Krawczyk, ed.), Lecture Notes in Computer Science, vol. 1462, Springer-Verlag, 1998, pp. 502–517.
15. Jacques Patarin, *Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt '88*, Advances in Cryptology – CRYPTO '95 (Vangalur S. Alagar and Maurice Nivat, eds.), Lecture Notes in Computer Science, vol. 963, Springer-Verlag, 1995, pp. 248–261.
16. Jacques Patarin, *Hidden Fields Equations (HFE) and Isomorphisms of Polynomials (IP): Two New Families of Asymmetric Algorithms*, Advances in Cryptology – EUROCRYPT 1996 (U. Maurer, ed.), Lecture Notes in Computer Science, vol. 1070, Springer-Verlag, 1996, pp. 33–48.
17. Jacques Patarin, Louis Goubin, and Nicolas Courtois, *Improved Algorithms for Isomorphisms of Polynomials*, Advances in Cryptology – EUROCRYPT '98 (Kaisa Nyberg, ed.), vol. 1403, 1998, pp. 184–200.
18. Douglas R. Stinson and Ruizhong Wei, *Combinatorial Properties and Constructions of Traceability Schemes and Frameproof Codes*, SIAM Journal on Discrete Mathematics **11** (1998), no. 1, 41–53.
19. *Specifications of SFLASH*, available from the site of the NESSIE workshop at `https://www.cosic.esat.kuleuven.ac.be/nessie/workshop/`.