J. Cryptology (2001) 14: 137–150
DOI: 10.1007/s00145-001-0001-z

Journal of
CRYPTOLOGY

Decomposing Attacks on Asymmetric Cryptography Based on Mapping Compositions

Dingfeng Ye
State Key Laboratory of Information Security,
Graduate School, Academia Sinica,
Beijing 100039-08, China
yedingfeng@hotmail.com

Zongduo Dai
State Key Laboratory of Information Security,
Graduate School, Academia Sinica,
Beijing 100039-08, China
yangdai@mimi.cnc.ac.cn

Kwok-Yan Lam
School of Computing,
National University of Singapore,
10 Kent Ridge, Singapore 119260
lamky@comp.nus.edu.sg

Communicated by Tom Berson

Received November 1998 and revised October 2000
Online publication 9 March 2001

Abstract. Given the algebraic expression of the composition of two mappings how can one identify the two components? This is the problem of mapping decomposition, of which the usual function-decomposition problem [8] is a special case. It was believed that this problem is intractable in general. Some public key cryptosystems (PKC) are based on the difficulty of this mathematical problem. Two types of such PKCs are FAPKC, proposed by Tao [16], and the "2R-schemes," proposed by Patarin and Goubin [11], [12]. FAPKC is based on composing finite automata (FA), while the "2R-schemes" use quadratic functions as the components. In this paper the decomposition problem for FA and for quadratic functions is investigated. Several methods for FA decomposing and one for quadratic functions are discovered. It is demonstrated that FA composition often exposes essential information about the components and that the full expression of composition of quadratic functions should not be given in 2R-schemes.

Key words. Finite automaton, Quadratic function, Algebraic expression, Composition, Decomposition, Public key cryptosystem.

1. Introduction

One approach to construct public key cryptosystems (PKC) makes use of mapping compositions, here mapping means any method to change inputs into outputs. The basic idea is as follows: a user chooses several easily invertible mappings which he keeps secret, computes the algebraic expression of their composition and makes it public; then anyone can do encryption or verify signatures using the public key, but will face a set of complicated algebraic equations when he tries to decrypt cipher texts or to forge signatures. Both FAPKC and "2R-schemes" make use of this idea, but the mappings (finite automata, FA) in FAPKC are sequential while those in the "2R-schemes" are blockwise. An obvious advantage of these PKCs is that the encrypting and signing can be made very efficient and be implemented with very simple hardware, this is very attractive for small detached devices such as smart cards.

FAPKC was proposed by Tao [16], [17], based on the so-called "weak invertibility theory of finite automata". In early designs of FAPKC, the public key is expressed by algebraic equations, and all published (in China) examples are broken [4], [1], [14]. In later work [15], [3] the public and private keys are only described in very coarse concepts without giving any explicit construction. Yet the scheme is broken because the suggested sizes of some parameters related to security are too small [7]. The attack given in [7] is a combination of message attack and a kind of trapdoor attack, it can be defended if the said parameter sizes are increased. Currently designs and implementations of FAPKCs are still under way, though no details of the construction have been made public. Our contribution in this work is that we give several methods for decomposing composed FAs which should be taken into account in designing new FAPKCs. These methods exploit some properties of FA composition which usual function composition does not have.

The 2R-schemes [11], [12] use composition of quadratic functions as public keys, based on the difficulty of the usual function-decomposition problem [8]. Biham gave an attack [2] if the components are constructed using S-boxes. Our attack, which was reported at CRYPTO '99 [18], treated the components as random quadratic functions. However, the formulation in [18] needs $q \geq 5$ which we now show is not essential. We give more evidences of the feasibility of the attack. We also discuss the potential of the approach when complete equations are not given. However, no efficient algorithm is found when we are given less than $n - 1$ components of the composition. This means that the 2R-schemes might be secure when some additional techniques [13] are exploited.

The rest of this paper is organized as follows: Section 2 defines the problem of mapping decomposition. Section 3 contains some preliminaries on FAs. Sections 4–6 describe the methods for decomposing FAs. Section 7 gives an example to illustrate FA decomposing methods. Section 8 is devoted to decomposition of quadratic functions.

2. The Problem of Mapping Decomposition

Unlike the problem of integer factorization, the problem of mapping decomposition is not even a well-defined mathematical problem. There are some ambiguities with a

general mention of the problem. This is because, for mappings, there are no analogies of "prime" and "uniqueness of decomposition". On the other hand, for a specific scheme of asymmetric cryptography based on mapping composition, this problem is usually well defined. In order to make our statements unequivocal, we need to define some terms.

Definition 1. A component model, or simply a model, is a set $\mathcal{M}$ of mappings, together with a probability distribution.

In this paper a component model should be understood as the set of all possible outputs of some process in generating private keys in a PKC based on mapping composition. Component models can be composed as follows:

Definition 2. Given two component models $\mathcal{M}_1$, $\mathcal{M}_2$, let

$$\mathcal{M}_1 \times \mathcal{M}_2 = \{f \circ g\colon\ f \in \mathcal{M}_1, g \in \mathcal{M}_2\}.$$

The problem of mapping decomposition can be stated as follows:

Problem 1. Let $\mathcal{M}_1$, $\mathcal{M}_2$ be two component models. Assume all mappings are expressed in some algebraic form. Given an element $h \in \mathcal{M}_1 \times \mathcal{M}_2$, find $f \in \mathcal{M}_1$, $g \in \mathcal{M}_2$ such that $h = f \circ g$.

3. Preliminaries on FAs

In this paper all FAs are the so-called input memory FAs which can be identified with functions of the following form:

$$f = f(t_{-h},\ t_{-h+1}, \ldots, t_0)\colon\ X^{h+1} \to X,$$

where we fix $X = F_q^l$, the l-dimensional vector space over a finite field F_q; and the t_{-i}'s are intermediate variables taking values from X. We call the subscripts of t_{-i}'s temporal indexes. Note that for FAPKC X is very small ($\leq 2^8$ elements).

The function f of this form defines a mapping from sequences to sequences of the same length as follows. Suppose $x_0x_1\cdots x_n$ is the input, then the output is $y_0y_1\cdots y_n$, where

$$y_i = f(x_{i-h}, x_{i-h+1}, \ldots, x_i)$$

and $x_{-h}x_{i-h+1}\cdots x_{-1}$ is the initial state. Corresponding to the composition of mappings, the composition rule for such functions is as follows. Let $f = f(t_{-h}, \ldots, t_0)$, $g = g(t_{-h'}, \ldots, t_0)$ be two FAs. Define the composition of f and g to be

$$f \circ g = f(g(t_{-h-h'}, \ldots, t_{-h}), \ldots, g(t_{-h'}, \ldots, t_0)). \tag{1}$$

An FA $f = f(t_{-h}, t_{-h+1}, \ldots, t_0)$ is called linear if f is an F_q-linear function; it is called τ-weakly invertible ($\tau \geq 0$ is an integer) if, given output sequence $y_0y_1\cdots y_\tau$ and the initial state $x_{-h}x_{i-h+1}\cdots x_{-1}$, x_0 can be uniquely determined.

Next, we introduce algebraic expressions of FAs. We denote by $M_{k,l}(R)$ the set of all $k \times l$ matrices over a ring R, by $M_{l,l}^*(R)$ the set of $l \times l$ matrices with nonzero determinant, and by $GL_l(R)$ the set of $l \times l$ invertible matrices, and we use $\mathcal{F}$ to denote the set of all FAs.

We denote by $\mathcal{F}_{1,l}$ the set of functions $f = f(t_{-h}, \ldots, t_0)$: $X^{h+1} \to F_q$, for all $h \geq 0$, where the t_{-i}'s are variables taking value from $X = F_q^l$. For any $f \in \mathcal{F}_{1,l}$ we can write

$$f = \sum_{\underline{a}} c_{\underline{a}} \underline{t}^{\underline{a}}, \tag{2}$$

where $\underline{a}$ runs through all subsets of $\{(-i, j, k)\colon\ 0 \leq i \leq h, 1 \leq j \leq l, 0 < k < q\}$, $c_{\underline{a}} \in F_q$, and

$$\underline{t}^{\underline{a}} = \prod_{(-i,j,k) \in \underline{a}} t_{-i,j}^k,$$

where $t_{-i,j}$ is the jth coordinate component of t_{-i}.

Equation (2) is called the algebraic expression of f. We define the start index of a monomial $\underline{t}^{\underline{a}}$ (or $\underline{a}$), denoted as $\mu(\underline{t}^{\underline{a}})$ (or $\mu(\underline{a})$), to be $\min\{i\colon\ (-i, j, k) \in \underline{a}\}$.

From now on, we let $R = F_q[Z]$, the polynomial ring of one variable Z over F_q, Z is actually the delay operator. We make $\mathcal{F}_{1,l}$ into an R-module by defining

$$Z^\mu \underline{t}^{\underline{a}} = \underline{t}^{\underline{a}^{(-\mu)}},$$

where $\underline{a}^{(-\mu)} = \{(-i - \mu, j, k)\colon\ (-i, j, k) \in \underline{a}\}$. We can rewrite (2) as $f = CT$, where $C \in M_{1,n}(R)$, T is the transpose of a vector $(T_1, \ldots, T_n)$, sometimes used as a set in this paper, and the T_i's are monomials with $\mu(T_i) = 0$. This kind of expression is called compact, and we call T the structure vector of f.

Similarly, any $f \in \mathcal{F}$ can be written as $f = C_f T$ with T being of the same form as above, and $C_f \in M_{l,n}(R)$. Linear FAs can be written as

$$L = C_L \begin{pmatrix} t_{0,1} \\ t_{0,2} \\ \vdots \\ t_{0,l} \end{pmatrix},$$

with $C_L \in M_{l,l}(R)$. It is easily seen that for any $f = C_f T \in \mathcal{F}$, $L \circ f = C_L C_f T$. So we can identify L with C_L. A linear FA L is weakly invertible if and only if $\det(C_L) \neq 0$.

Following is some notation used in this paper:

- For any matrix A over R, define the degree of A, denoted by $\deg(A)$, to be the maximal degree of its entries; matrices over F_q are called constant matrices.
- For any $f \in \mathcal{F}_{1,l}$, let $\mathrm{ndeg}(f)$ denote the nonlinear degree of f, which is the maximal value among the degrees of the monomials appearing in the algebraic expression of f. For $f \in \mathcal{F}$, $\mathrm{ndeg}(f)$ is the maximal value among the nonlinear degrees of its components.
- For each monomial $\tilde{T} = \underline{t}^{\underline{a}}$, define its memory order as $m(\tilde{T}) = \max\{i\colon\ (-i, j, k) \in \underline{a}\}$, define its span as $s(\tilde{T}) = m(\tilde{T}) - \mu(\tilde{T})$.

- For any $f \in \mathcal{F}_{1,l}$, let $m(f)$ denote its memory order, which is the maximal value among the memory orders of the monomials appearing in the algebraic expression of f.

Next, we mention the general constructing methods of FAs whose weak inverse can be routinely formulated. τ-weakly invertible linear FAs and 0-weakly invertible FAs are the only known two basic classes of FAs whose structure is clear. A process for constructing τ-weakly invertible FAs may take the following measures:

- Primitives: constructing τ-weakly invertible linear FAs or 0-weakly invertible FAs using methods in [7].
- Composition: the composition of a τ_1-weakly invertible FA and a τ_2-weakly invertible FA is a $(\tau_1 + \tau_2)$-weakly invertible FA.
- Summing: if f is τ-weakly invertible, then so is $f + Z^{1+\tau}g$ for any g.

Now we consider the models for the decomposition problem of FAs. There are two basic types of models: a linear model which consists of only linear FAs and a 0-model which consists of nonlinear 0-weakly invertible FAs. In this paper we consider only the compositions of these two kinds of models; we do not take into consideration the summing method for constructing FAs. So the models considered in this paper are products in which linear models and 0-models appear alternatively. We make the following conventions. All models are stable under linear automorphism, that is, if $f \in \mathcal{M}$, then for any $G \in GL_l(F_q)$, we have $f \circ G \in \mathcal{M}$ and $G \circ f \in \mathcal{M}$. We also write $f \sim h$, if $h = f \circ G$ or $h = G \circ f$ for some $G \in GL_l(F_q)$; similar notation applies for matrices. Another convention is that all elements in a model should have the same type of algebraic form. Thus, when we say we are given a model $\mathcal{M}$, it means that we are given a structure vector T and a parameter domain $C(\mathcal{M}) \subset M_{l,n}(R)$, such that the elements of $\mathcal{M}$ are exactly $\{CT\colon\ C \in C(\mathcal{M})\}$. Some properties of $C(\mathcal{M})$ may also be assumed to be known, for example, we may assume that the distribution of degrees at each column is known.

4. Decomposing from Outside

In this section we demonstrate how to attack the decomposition problem for model types "LN" and "PM". Here "L" stands for linear, "N" stands for nonlinear, "P" stands for permutation, and "M" stands for a general model. In other words, a composed model $\mathcal{M}_1 \times \mathcal{M}_2$ is said to be of type "LN" if $\mathcal{M}_1$ is linear and $\mathcal{M}_2$ is nonlinear; the type "PM" should be interpreted similarly.

4.1. *Decomposing "LN"*

In this subsection we are given a composed model $\mathcal{L} \times \mathcal{N}$, where $\mathcal{L}$ is linear and $\mathcal{N}$ is nonlinear. We assume $\mathcal{N}$ satisfies: for a random $N \in \mathcal{N}$, it is very likely that there is no non-constant matrix A such that $A \circ N \in \mathcal{N}$.

Given the algebraic expression of an element $f = B \circ N \in \mathcal{L} \times \mathcal{N}$, where N has the property stated above, we wish to find $B' \in \mathcal{L}$ and $N' \in \mathcal{N}$, such that $f = B' \circ N'$. Such B', N' are unique in the sense that $B \sim B'$ and $N \sim N'$. Let $f = C_f T$ be the algebraic

expression. We must have $N = CT$, and $C_f = BC$, for some $C \in M_{l,n}(R)$. Translated to matrix terminology, our problem can be stated as follows:

Problem 2. Given a parameter domain $C(\mathcal{N}) \subset M_{l,n}(R)$ and a $C \in C(\mathcal{N})$, with the property that

$$\{A \in M_{l,l}(K):\ AC \in C(\mathcal{N})\} \subseteq M_{l,l}(F_q),$$

where K is the fraction field of R; and given $C_f = BC$ with $B \in M^*_{l,l}(R)$, find a $B' \sim B$.

This problem can be solved by the following approach: Firstly choose l columns of C_f such that they form a submatrix A of C_f with $\det(A) \neq 0$. Compute $C' = A^{-1}C_f$, which is a matrix over $F_q(Z)$, the fraction field of R. Let A' be the corresponding submatrix of C; we have $C' = A'^{-1}C$, so C' is irrelevant to B. Next try to solve the matrix equation:

$$A_x C' \in C(\mathcal{N}), \tag{3}$$

where $A_x \in M_{l,l}(R)$ is the unknown matrix. Suppose we can find a solution A_x of (3), then we can see that $AA_x^{-1} \sim B$ and we are done.

However, the condition "$\in C(\mathcal{N})$" in (3) can hardly be expressed in linear or algebraic form in practice. To circumvent this we can replace it with weaker conditions under which the solution of (3) remains unchanged. Such weaker conditions may vary with specific construction of $\mathcal{N}$. In many cases, we believe that considering only the degrees of columns of $C(\mathcal{N})$ is enough. Suppose the ith column of $C(\mathcal{N})$ has expected degree d_i, then we can expect that the linear space generated by rows of a solution of (3) is the solution space of the following system of linear equations:

$$\underline{x}\beta_i \in R \quad \text{and} \quad \deg(\underline{x}\beta_i) \leq d_i, \tag{4}$$

where $\underline{x}$ is the unknown taking value from $M_{1,l}(R)$, and β_i, $1 \leq i \leq n$, is the ith column of C'.

We outline the procedures in solving Problem 2 as follows:

Step 1. Choose l columns of C_f such that they form a submatrix A of C_f with $det(A) \neq 0$ and the corresponding d_i's are as small as possible.

Step 2. Compute $C' = A^{-1}C_f$, which is a matrix over $F_q(Z)$, the fractional field of R.

Step 3. Solve (4). Let V be the solution space. If V has dimension exceeding l, return to Step 1 and proceed with other choices of the columns.

Step 4. Choose any basis of V to form a matrix A'', then $B' = AA''^{-1}$ is what we want.

Remark 1. Equation (4) can be translated into a system of linear equations over F_q, with the number of unknowns equal to $(1 + d)l$ where d is the maximal value among the d_i's chosen in Step 1.

4.2. *Decomposing "PM"*

Now we are given a composed model $\mathcal{P} \times \mathcal{M}$ of type "PM", and the algebraic expression of an instance $f = P \circ M$ of this model, where P is a nonlinear permutation on the

input space X. Suppose $P^{-1} = T_b Q$ where T_b is a translation and Q is a permutation which maps $\underline{0}$ to $\underline{0}$. Our task is to find a $Q' \sim Q$. This is the same as determining $\mathcal{L}(Q)$, the linear space generated by coordinate components of Q.

For any function $\sigma\colon F_q^m \to F_q^{m'}$, let $T(\sigma)$ denote the set of monomials appearing in the algebraic expression of σ. Our method is based on the following observation. Let $\sigma\colon F_q^m \to F_q^l$ be a random function with $\text{ndeg}(\sigma) \le k$, where $m \ge l$, $k << m(q-1)$. Then in most cases, if $\lambda\colon F_q^l \to F_q$ makes $T(\lambda \circ \sigma) \subset T(\sigma)$, then λ must be an affine function, i.e. "a linear function + a constant".

Let $T^{(0)} = (T_1^{(0)}, T_2^{(0)}, \ldots, T_{q^l-2}^{(0)})$ (recall that q^l is small) be all monomials over t_0 except the two trivial ones: 1 and $\prod_{1\le i\le l} t_{0,i}^{q-1}$. Then there exists $A \in M_{l,q^l-2}(F_q)$ such that $Q = AT^{(0)}$. For our purpose, it is enough to find the linear F_q-space $\mathcal{L}(A)$. Suppose $T^{(0)} f = C\hat{T}$, where $C = (C_1, C_2, \ldots, C_n) \in M_{q^l-2,n}(F_q)$ and $\hat{T} = T(T^{(0)} f) = (\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_n)$. Define a linear subspace V in $M_{1,q^l-2}(F_q)$ as

$$V = \{\underline{x} \in M_{1,q^l-2}(F) | \underline{x} C_j = 0, \hat{T}_j \notin T(M), 1 \le j \le n\}.$$

It is clear that $\mathcal{L}(A) \subset V$. By the observation made in the previous paragraph, we may expect in most cases that $\mathcal{L}(A) = V$.

In order to find V, it is enough to find $q^l - 2 - l$ C_j's which are linearly independent and the corresponding $\hat{T}_j$ are not in $T(M)$. Let $k = \text{ndeg}(\hat{T}_j)$, then the number of monomials dividing $\hat{T}_j$ is bounded by 2^k. A multiplication of two polynomials with 2^k terms needs at most $O(2^{2k})$ operations in F_q. To compute one C_j, at most q^l such multiplications are needed. To determine V, we need to compute $O(q^l)$ such C_j's. So the total computational complexity for determining V is bounded by $O((q^l 2^K)^2)$, no matter how complex f is, where $K = \max_{\text{such } j}\{\text{ndeg}(\hat{T}_j)\}$ ($K = \text{ndeg}(M) + 1$ is often enough).

5. Decomposing from Inside

When decomposing from inside, the objective is to determine the linear space generated by the components of the inner mapping (FA or usual function). In general, it is not easy to obtain this linear space directly, but some space related to it can be obtained using the techniques described in subsequent sections. Under certain circumstances, the linear space we wanted can be derived from this related space. To get such a space, formal partial differentials are useful tools, so we start with a brief introduction to formal partial differentials.

5.1. *Formal Partial Differentials*

Let K be any ring, and let $x_1, x_2, \ldots, x_n$ be n independent variables. Define partial differentials of the first order, $\partial/\partial x_i$, $i = 1, 2, \ldots, n$ as follows: $\partial/\partial x_i\colon K[x_1, x_2, \ldots, x_n] \to K[x_1, x_2, \ldots, x_n]$ is a $K[x_1, \ldots, \hat{x_i}, \ldots, x_n]$-module homomorphism satisfying

$$\frac{\partial x_i^k}{\partial x_i} = k x_i^{k-1},$$

where the "hat" stands for "omitted". It is not hard to verify that, as operators on $K[x_1, x_2, \ldots, x_n]$, the $\partial/\partial x_i$'s commute with each other. Now for any monomial $\prod x_{i_j}^{k_j}$

with degree $k = \sum k_j$, define inductively

$$\frac{\partial}{\partial \prod x_{i_j}^{k_j}} = \frac{\partial}{\partial x_{i_1}} \left(\frac{\partial}{\partial (\prod x_{i_j}^{k_j})/x_{i_1}} \right).$$

The most important property of partial differentials is the following:

$$\frac{\partial fg}{\partial x_i} = \frac{\partial f}{\partial x_i} g + f \frac{\partial g}{\partial x_i}.$$

Using this, and by induction on k, it is not hard to prove

Lemma 1. *Let $L_1, L_2, \ldots, L_k$ be k linear expressions, let $\tilde{T}$ be any monomial of degree $k-1$, then $(\partial \prod L_i)/\partial \tilde{T}$ is a K-linear combination of $L_1, L_2, \ldots, L_k$.*

5.2. *Decomposing "NL"*

In this subsection we are given a composed model $\mathcal{N} \times \mathcal{L}$, where $\mathcal{L}$ is linear and $\mathcal{N}$ is nonlinear. Given the algebraic expression of an element $f = N \circ B \in \mathcal{N} \times \mathcal{L}$, our objective is to find a $B' \sim B$, i.e. to find the vector space $\mathcal{L}(B)$ generated by rows of B. Let $\mathcal{L}^{(k)} = \bigoplus_{0 \le i \le k} Z^i \mathcal{L}(B)$.

Lemma 2. *Let $k = \mathrm{ndeg}(N) = \mathrm{ndeg}(f)$, suppose all monomials of degree k appearing in the algebraic expression of N have spans no larger than d. Then for any monomial $\tilde{T} = \underline{t}^{\underline{a}}$ of degree $k-1$, we have $\partial f_i/\partial \tilde{T} \in \mathcal{L}^{(k')}$, where f_i is a coordinate component of f, $k' = d + \mu(\tilde{T})$ and the constant term is neglected.*

Proof. f_i can be written as the form $\sum \prod_{1 \le i \le k} Z^{n_i} L_i +$ lower terms, where $L_i \in \mathcal{L}(B)$, $n_1 \le n_2 \le \cdots \le n_k$. We know that $n_k - n_1 \le d$. A term $\prod_{1 \le i \le k} Z^{n_i} L_i$ can contribute to $\partial f_i/\partial \tilde{T}$ only if $n_1 \le \mu(\tilde{T})$. Thus

$$\frac{\partial f_i}{\partial \tilde{T}} = \sum_{n_1 \le \mu(\tilde{T})} \frac{\partial \prod_{1 \le i \le k} Z^{n_i} L_i}{\partial \tilde{T}}$$

and the lemma follows from Lemma 1. □

Let V_μ denote the linear space generated by all $\partial f_i/\partial \tilde{T}$'s as in the above lemma with $\mu(\tilde{T}) \le \mu$. By adding V_{i-1} to V_i we get a sequence of vector spaces

$$V_0 \subseteq V_1 \subseteq V_2 \subseteq \cdots, \qquad V_i \subseteq \mathcal{L}^{(i+d)}.$$

Given such a sequence of vector spaces, $\mathcal{L}(B)$ might be recovered in many cases with various methods. The following gives one such method.

Suppose for some $i \ge 0$ we have $V_i = V_{i+1} \cap \mathcal{L}^{(i+d)}$ and $V_i \cap Z^{i+d+1} M_{1,l}(R) = 0$, then we have $V_{i+1} \cap Z^{i+d+1} M_{1,l}(R) \subseteq Z^{i+d+1} \mathcal{L}(B)$ and thus we get a subspace of $\mathcal{L}(B)$. In this way $\mathcal{L}(B)$ could hopefully be recovered.

In general, let

$$U = \sum_{i\geq 0} \frac{Z^{i+1}M_{1,l}(R)\cap \mathcal{L}^{(i)}}{Z^{i+1}}, \qquad U_i = \frac{V_{i+1}\cap Z^{i+d+1}M_{1,l}(R)}{Z^{i+d+1}},$$

then we have $W = \sum_i U_i \subseteq U + \mathcal{L}(B)$. Note that in most cases $\dim(U)$ is small and $\mathcal{L}(B) \subseteq W$, thus we get a small superspace of $\mathcal{L}(B)$. It is possible to fix $\mathcal{L}(B)$ further by other considerations.

5.3. *Decomposing "MP"*

Given a composed model $\mathcal{M}\times\mathcal{P}$, where $\mathcal{P}$ is a permutation model, and given the algebraic expression of an element $f = M\circ P$, our objective is to find the vector space $\mathcal{L}(P)$ generated by coordinate components of P. This turns out to be an easy problem, and we can solve it even without knowledge of the component models.

Let $n = \mathrm{ndeg}(f), n_0 = \mathrm{ndeg}(M), n_1 = \mathrm{ndeg}(P)$. Let S be the set of monomials appearing in the algebraic expression of P. Let $f = CT$ be the compact algebraic expression of f. The following describes how to obtain n_0, n_1, S solely from T.

Each element T_j of T can be uniquely written as $\prod_i Z^{n_i}T_i^{(0)}$, where $n_1 < n_2 < \cdots$, each $T_i^{(0)}\in T^{(0)}$, the set of monomials containing only components of t_0. We say these $T_i^{(0)}$'s belong to T_j, written as $T_i^{(0)}\perp T_j$, and we denote the number of these i's as $n(T_j)$. Let $T^0 = \{T_i\in T\colon \mathrm{ndeg}(T_i) = n\}$, $S_0 = \{T_i^{(0)}\in T^{(0)}\colon T_i^{(0)}\perp T_j \text{ for some } T_j\in T^0\}$. It is easy to understand that in usual cases, we have $n_1 = \min\{\mathrm{ndeg}(T_i^{(0)})\colon T_i^{(0)}\in S_0\}$, $n_0 = n/n_1$. Now let $T^* = \{T_j\in T\colon n(T_j) = n_0\}$, then in most cases we have

$$S = \{T_i^{(0)}\in T^{(0)}\colon T_i^{(0)}\perp T_j \text{ for some } T_j\in T^*\}.$$

Let C^* be the submatrix of C corresponding to T^*, and let $f^* = C^*T^*$, f_k^* be a coordinate component of f^*. For any monomial $\tilde{T}$ of the form $\tilde{T} = \prod_{1\leq i\leq n_0-1} Z^{n_i'}T_i$, where $\mathrm{ndeg}(T_i) = n_1$ for all i, and $n_0' < n_1' < \cdots$. Then we must have:

Lemma 3. *Notation as above, let*

$$f_k^* = \tilde{T}C(f_k^*,\tilde{T}) + \textit{ terms not divisible by } \tilde{T}.$$

Then $C(f_k^*,\tilde{T})\in\bigoplus_{i\geq 0} Z^i\mathcal{L}(P)$.

Proof. It is easily seen that f_k^* can be written as the form $\sum\prod_{1\leq i\leq n_0} Z^{n_i}L_i$, where $L_i\in\mathcal{L}(P)$, $n_1 < n_2\cdots < n_k$. A term $\prod_{1\leq i\leq n_0} Z^{n_i}L_i$ can contribute to $C(f_k^*,\tilde{T})$ if and only if there exists $1\leq j\leq k$, such that $(n_1,\ldots,\hat{n_j},\ldots,n_{n_0}) = (n_1',\ldots,n_{n_0-1}')$ and each T_i is a term of the corresponding $L_{i'}$, in which case it contributes a scalar multiple of $Z^{n_j}L_j$. □

Given any element of $\sum_{i\geq 0}\mathcal{L}(P)$, its coefficient of each Z^i lies in $\mathcal{L}(P)$. Thus some elements of $\mathcal{L}(P)$ can be easily obtained in this way, very likely they will generate $\mathcal{L}(P)$.

6. Decomposing "MM"

In this section we are given a composed model $\mathcal{M}_1 \times \mathcal{M}_2$ in which the two component models are general. Given an element $f = M_1 \circ M_2$ of this model, our objective is to find the linear space $\mathcal{L}(M_2)$, the linear space generated by components of M_2. This can be done if M_1 has certain properties with respect to M_2 which are described below. For ease of description, we assume $\mathrm{ndeg}(M_1) = 2$, though the method is generally applicable.

Let $f = C_f T$, $M_2 = C'T'$ be the compact algebraic expressions of f and M_2, respectively. Let $m = m(M_2)$. Let $T^* = \bigcup_i Z^i T'$. For any $\tilde{T} \in T^*$, define

$$E(\tilde{T}) = \{\tilde{T}' \in T^* \colon \exists T_1, T_2 \in T^* \text{ such that } T_1 T_2 = \tilde{T}\tilde{T}'\}.$$

Similar to Lemma 3, we have

Lemma 4. *Notation as above, for any $\tilde{T} \in T^*$, let*

$$f_k = \tilde{T}(C(f_k, \tilde{T}) + \textit{ some terms not in } T^*) + \textit{ terms not divisible by } \tilde{T},$$

where $C(f_k, \tilde{T})$ has only terms in T^*. Then there exists a linear combination Error *of elements in $E(\tilde{T})$, such that*

$$C(f_k, \tilde{T}) - \text{Error} \in V = \bigoplus_{0 \le i \le m(M_1)} Z^i \mathcal{L}(M_2),$$

where f_k is a coordinate component of f.

The above lemma says that, up to some error terms, we can obtain elements of V. Note that the temporal indexes of elements in $E(\tilde{T})$ are in the range $[\mu(\tilde{T}) - m, m(\tilde{T}) + m]$. Assume that all non-zero elements of $\mathcal{L}(M_2)$ have memory order m (which is true in most cases), then under suitable conditions described below, $\mathcal{L}(M_2)$ can be recovered.

The first method: Let V' be the linear space generated by all $C(f_k, \tilde{T})$'s in the above lemma with $\mu(\tilde{T}) \le \mu$ for some fixed μ, and by neglecting all terms with memory order no less than $\mu - m$. If $m(M_1) >> 2m$ and there are enough terms of M_1 of span $> m$, then for suitable $\mu >> 2m$, we can expect that the subspace of V' consisting of elements with memory order not exceeding m is exactly $\mathcal{L}(M_2)$ with very high probability. Again, this subspace can be obtained by Gaussian elimination.

The second method: Let V'' be the linear space generated by all $C(f_k, \tilde{T}) -$ Errors in above lemma. Suppose we can obtain a subspace V''' of V'' such that $\mathcal{L}(M_2) \subset V'''$, then $\mathcal{L}(M_2)$ can be obtained by Gaussian elimination. One could obtain such a V''' from the $C(f_k, \tilde{T})$'s by repairing the error terms using linear algebra.

7. An Example

Up to now, the most sophisticated publicly known examples of FAPKC are "quadratic form FAPKC's [3]. There are two forms: one is of type "LPL" and the other of type "LPLP", where the permutation is a fixed exponentiation in the finite field F_{q^l} of algebraic degree 2. For the toy examples of these two schemes given in [3], we have decomposed

the public keys by hand computing. One example of "LPL" is $f = C_f T$, where $C_f = (C_1, C_2)$,

$$C_1 = \begin{pmatrix} z^2 & 1+z^2 & 1+z & 0 & z^2 & z \\ 1 & z^2 & 1 & 0 & z+z^2 & z^2 \\ 1+z^2 & 1+z & z+z^2 & z+z^2 & z^2 & z+z^2 \end{pmatrix},$$

$$C_2 = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1+z & 0 & 0 & z \\ 0 & 0 & z & 0 & 0 & 1 & 0 & 0 & 1+z \\ z & 0 & 1+z & z & z & 0 & 0 & z & 1+z \end{pmatrix},$$

and

$$T = \begin{pmatrix} T^{(1)} \\ T^{(2)} \end{pmatrix},$$

where $T^{(1)} = (t_{0,1}, t_{0,2}, t_{0,3}, t_{0,1}t_{0,2}, t_{0,1}t_{0,3}, t_{0,2}t_{0,3})^t$ and $T^{(2)} = (t_{0,1}t_{-1,1}, t_{0,1}t_{-1,2}, t_{0,1}t_{-1,3}, t_{0,2}t_{-1,1}, t_{0,2}t_{-1,2}, t_{0,2}t_{-1,3}, t_{0,3}t_{-1,1}, t_{0,3}t_{-1,2}, t_{0,3}t_{-1,3})^t$.

Corresponding to the terms $(t_{0,1}t_{-1,1}, t_{0,1}t_{-1,3}, t_{0,2}t_{-1,3})$, we get a submatrix of C_f,

$$A = \begin{pmatrix} 0 & 1 & 1+z \\ 0 & z & 1 \\ z & 1+z & 0 \end{pmatrix}, \qquad \det(A) \neq 0,$$

$C' = A^{-1}C_f$ equals

$$\begin{pmatrix} 1+z & 1+z & 1+z & 1+z & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1+z & 0 & 0 & z & z & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1+z & z & 1 & 0 & z & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

It is actually in the "PL" model, and the rest of the "LN" decomposing method is not needed.

Next we decompose $C'T$ using either the "PM" or the "NL" method. The C_j's as in the "PM" method corresponding to the terms $t_{-1,1}t_{-1,2}, t_{-1,1}t_{-1,3}, t_{-1,2}t_{-1,3}$ form the following matrix,

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}^t,$$

which has rank 3. Solving the equation $\underline{x}C_j = 0$ we get P^{-1}:

$$(t_1 + t_2 + t_3 + t_2t_3, t_2 + t_1t_3, t_2 + t_3 + t_1t_2).$$

To see how the "NL" method works, note that "P" has span 0, the inner linear FA is simply obtained by the partial derivatives of the first component of $C'T$ with respect to $t_{0,1}, t_{0,2}, t_{0,3}$:

$$\begin{pmatrix} z & 1 & 1 \\ 1+z & z & 0 \\ 1 & z & 0 \end{pmatrix}.$$

8. Quadratic Function Decomposition

Let $f, g\colon F_q^n \to F_q^n$ be two quadratic functions. Let $h = f \circ g$ and let $h_i,\ f_i, g_i$ denote components of $h,\ f, g$, respectively. Given the algebraic expression of h, i.e. each h_i is given as a polynomial in $F_q[x_1, x_2, \ldots, x_n]$, we wish to find f', g' so that there exists an affine permutation A such that $f' = fA^{-1}, g' = Ag$. This is equivalent to finding the vector space generated by components of g together with 1. The difficult part is to determine the terms of degree 2, because the linear and constant terms can easily be determined by solving linear equations after the degree 2 terms of f', g' are determined.

From now on we assume f, g to be homogeneous, i.e. f_i, g_i are quadratic forms. The decomposition problem is the same as deciding $\mathcal{L}(g)$, the linear space generated g_i, $1 \le i \le n$. When $q \le 4$, the components of $h = f \circ g$ as reduced polynomials (the degree of any variable is $< q$) may not be homogeneous. We ignore all terms of lower degree in this case. Moreover, we make a convention: all terms of lower degree in the reduced form of the result of a polynomial operation will be ignored.

Let $\mathcal{L}$ denote the linear space of linear forms, let $\mathcal{Q}$ denote the linear space of all quadratic forms, and let $V(g) = \sum_{1\le i\le n} X_i\mathcal{L}(g)$. For any subspace $\mathcal{L}'$ of $\mathcal{L}$ and any linear space W of homogeneous polynomials of degree 3, define

$$(W : \mathcal{L}') \stackrel{\text{def}}{=} \{r \in \mathcal{Q}\colon r\mathcal{L}' \subseteq W\}.$$

When $\mathcal{L}'$ has dimension 1, say, generated by F, we also write $(W : \mathcal{L}')$ as $(W : F)$.

Let $\tilde{\mathcal{L}}(g) = (V(g) : \mathcal{L})$, and $\tilde{n} = \dim(\tilde{\mathcal{L}}(g))$. Obviously, we have $\mathcal{L}(g) \subseteq \tilde{\mathcal{L}}(g)$. By our method in the following, we can only get $\tilde{\mathcal{L}}(g)$, so we need the equality to hold to get $\mathcal{L}(g)$. It should be noted that this is not necessary in decomposing $2R$-schemes, though it is often satisfied. As long as we can get $\tilde{\mathcal{L}}(g)$, we can get a decomposition $h = \tilde{f} \circ \tilde{g}$ where $\tilde{f}\colon F_q^{\tilde{n}} \to F_q^n, \tilde{g}\colon F_q^n \to F_q^{\tilde{n}}$. The techniques for attacking one round schemes in [12] might be applied to $\tilde{f}$ or $\tilde{g}$ to identify f, g further, when $\tilde{n} - n$ is small.

Now let V denote the vector space generated by $\partial h_j/\partial X_i$, for all i, j.

Lemma 5. $V \subseteq V(g)$.

Proof. We can write h_j in the form $\sum a_{k,l}g_kg_l$ according to our convention, so we have

$$\frac{\partial h_j}{\partial X_i} = \sum a_{k,l}\left(\frac{\partial g_k}{\partial X_i}g_l + \frac{\partial g_l}{\partial X_i}g_k\right) \in V(g),$$

which completes the proof. □

If $V = V(g)$, we can obtain $\tilde{\mathcal{L}}(g)$ by computing $(V : \mathcal{L})$. Otherwise, if $\delta = \dim(V(g)) - \dim(V) < \tilde{n}$, we might recover $V(g)$ as follows.

An algorithm to recover $V(g)$:

Step 1. Compute $(V : F)$ for sufficiently many random $F \in \mathcal{L}$, and choose one F such that $\dim(V : F)$ is minimal.

Step 2. Replace V with $V + (V : F)\mathcal{L}$, and return to Step 1.
Step 3. If the operation in Step 2 cannot enlarge V for many times, then output V.

Algorithm End

At the start of the algorithm we have $F(V : F) = V \cap F\mathcal{Q} \supseteq V \cap F\tilde{\mathcal{L}}(g)$ and thus

$$\dim((V : F)) \geq \dim(V) + \dim(F\tilde{\mathcal{L}}(g)) - \dim(V(g)) = \tilde{n} - \delta.$$

The equality holds with non-negligible probability by assuming $F\tilde{\mathcal{L}}(g)$ is a random subspace of $V(g)$ of dimension $\tilde{n}$. Thus the selected F as in Step 1 satisfies this equality. This equality implies $F(V : F) = V \cap F\tilde{\mathcal{L}}(g)$, i.e., $(V : F) \subseteq \tilde{\mathcal{L}}(g)$, and hence the updated V in Step 2 is still contained in $V(g)$. If $V \neq V(g)$, the operation in Step 2 will enlarge V with non-negligible probability, so the output of Step 3 should be $V(g)$. This proves the correctness of the above algorithm.

The condition $\delta < \tilde{n}$ is not so serious since it holds with probability approximately $1 - q^{-(\tilde{n}-1)^2}$ if the partial derivatives are regarded as random and independent. So the decomposition problem of quadratic functions is not hard as long as $\mathcal{L}(g) = \tilde{\mathcal{L}}(g)$. We conjecture this holds for the majority of g when $n \geq 3$.

These observations indicate that $2R$-schemes are dangerous if full expression of $h = f \circ g$ is made public. One can circumvent the above attack by exploiting additional techniques, as in [13]. One variation is $2R^-$, which does not publish the full expression of h. If we are given $n - 1$ components of h, then V has expected degree close to $n(n-1)$, and $(V : F)$ is non-trivial with probability greater than $q^{-1-n(n-1)+\dim(V)}$. Since $(V : F) \subseteq \tilde{\mathcal{L}}(g)$ with nontrivial probability, the scheme is still vulnerable. This suggests we should delete at least two components of h in $2R^-$. When given less than $n - 1$ components of h, our method will fail since the probability that $(V : F)$ is non-trivial is negligible.

9. Conclusion

We have given some trapdoor attacks on two public key schemes based on the idea of composing mappings: FAPKC and $2R$. For $2R$, our attack is effective if and only if at least $n-1$ equations of the composition mapping are exposed in the public key. It can be easily defeated by techniques of [13]. For FAPKC, our attacks are effective with respect to all publicly known examples and some conceivable constructions. It is a challenging problem to design FAPKC algorithms which are both practical and secure against the attacks in this paper and [7].

Acknowledgments

Thanks to Professor Yang Junhui for pointing out that the condition $q \geq 5$ is not essential in $2R$-decomposing. Thanks to Dr. Thomas A. Berson and the anonymous referee for valuable remarks on revising the manuscript of this work.

References

[1] F. Bao and Y. Igarashi, Break finite automata public key cryptosystem, in *Automata, Languages and Programming*, Lecture Notes in Computer Sciences, Vol. 944, Springer-Verlag, Berlin, 1995, pp. 147–158.

[2] E. Biham, Cryptanalysis of Patarin's 2-round public key system with S boxes, *Proceedings of CRYPTO* '99, Springer-Verlag, New York.

[3] X.M. Chen, The Invertibility Theory and Application of Quadratic Finite Automata, Doctoral thesis, Laboratory for Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing 100080, November 1996.

[4] D.W. Dai, K. Wu and H.G. Zhang, Cryptanalysis on a finite automaton public key cryptosystem, *Sci. China*, 1994 (in Chinese).

[5] Z.D. Dai and D.F. Ye, Weak invertibility of linear finite automata over commutative rings—classification and enumeration, *Kexue Tongbao* (Bulletin of Science), Vol. 4, No. 15 (1995), pp. 1357–1360 (in Chinese).

[6] Z.D. Dai and D.F. Ye, Weak invertibility of linear finite automata I—classification and enumeration of transfer functions, *Sci. China* (*Ser. A*), Vol. 39, No. 6 (1996), pp. 613–623.

[7] Z. Dai, D. Ye and K. Lam, Weak invertibility of finite automata and cryptanalysis of FAPKC, *Proceedings of ASIACRYPT* '98, Lecture Notes in Computer Science, Vol. 1514, Springer-Verlag, Berlin, 1998.

[8] M. Dickerson, The Functional Decomposition of Polynomials, Ph.D Thesis, TR89-1023, Deptartment of Computer Science, Cornell University, Ithaca, NY, July 1989.

[9] W. Diffie and M.E. Hellman, New directions in cryptograghy, *IEEE Trans. Inform. Theory*, Vol. IT-22, No. 6 (1976), pp. 644–654.

[10] J. Patarin, Cryptanalysis of the Matsumoto and Imai public key scheme of EUROCRYPT '88, *Advances in Cryptology, Proceedings of CRYPTO* '95, 1995, pp. 248–261.

[11] J. Patarin and L. Goubin, Trapdoor one-way permutations and multivariate polynomials, *Proceedings of ICICS* '97, Lecture Notes in Computer Science, Vol. 1334, Springer-Verlag, Berlin, 1997.

[12] J. Patarin and L. Goubin, Asymmetric cryptography with S-boxes, *Proceedings of ICICS* '97, Lecture Notes in Computer Science, Vol. 1334, Springer-Verlag, 1997.

[13] J. Patarin, L. Goubin and N. Courtois, C_{-+} and HM: variations around two schemes of T. Matsumoto and H. Imai, *Advances in Cryptology, Proceedings of ASIACRYPT* '98, Lecture Notes in Computer Science, Vol. 1514, Springer-Verlag, Berlin, 1998.

[14] Z.P. Qin and H.G. Zhang, Cryptanalysis of finite automaton public key cryptosystems, *Proceedings of Chinacrypt* '96, Science Press, Beijing, pp. 75–86. (in Chinese).

[15] R.J. Tao, On finite automaton one-key cryptosystem, in *Fast Software Encryption*, Lecture Notes in Computer Science, Vol. 809, Springer-Verlag, Berlin, 1993.

[16] R.J. Tao and S.H. Chen, A finite automaton public key cryptosystem and digital signatures, *Chinese J. Comput.*, Vol. 8 1985, pp. 401–409 (in Chinese).

[17] R.J. Tao and S.H. Chen, Two varieties of finite automaton public key cryptosystem and digital signatures, *J. Comput. Sci. Tech.*, Vol. 1 (1986), pp. 9–18.

[18] D.F. Ye, Z.D. Dai and K.Y. Lam, Cryptanalysis of 2R schemes, *Proceedings of CRYPTO* '99, Springer-Verlag, New York.