

Chapter 2

MATSUMOTO-IMAI CRYPTOSYSTEMS

In the previous chapter we discussed some early attempts to build MPKCs. However, these attempts were not very successful and it became very clear that new mathematical ideas were needed. The first such new idea was proposed by Matsumoto and Imai [Matsumoto and Imai, 1988]. Their key idea was to utilize both the vector space and the hidden field structure of k^n , where k is a finite field. More specifically, instead of searching for invertible maps over the vector space k^n directly, they looked for invertible maps on a field K , a degree n field extension of k , which can also be identified as an n dimensional vector space over k . This map could then be transformed into an invertible map over k^n .

One such cryptosystem, known as C^* or MI, attracted a lot of attention due to its high efficiency and potential use in practical applications. In fact, the MI cryptosystem was submitted as a candidate for security standards of the Japanese government. However, before the final selection, MI was broken by Jacques Patarin using an algebraic attack that utilizes linearization equations [Patarin, 1995]. This method takes advantage of certain specific hidden algebraic structures in MI.

Normally one would conclude that this is the end of MI, though in fact the subsequent story goes into the opposite direction. One reason is that the MI cryptosystem represents a fundamental breakthrough on the conceptual level in that it brought a totally new mathematical idea into the field and consequentially was widely explored and extended. Another reason is that there are many new variants of the MI cryptosystems that seem to have great potential, including the Sflash signature scheme [Akkar et al., 2003; Patarin et al., 2001], which was accepted in 2004 as one of the final selections for the New European Schemes for Signatures,

Integrity, and Encryption project [NESSIE, 1999] for use in low cost smart cards.

Indeed, the work of Matsumoto and Imai has played a critical role as a catalyst in this new area and has stimulated the subsequent development. In this chapter, we will present the MI cryptosystem in detail, Patarin's cryptanalysis of MI, the Plus-Minus variants, related attacks and security analysis.

2.1 Construction of a Matsumoto-Imai System

Let k be a finite field of characteristic two and cardinality q , and take $g(x) \in k[x]$ to be any irreducible polynomial of degree n . Define the field $K = k[x]/g(x)$, a degree n extension of k . In general the $\text{char}(k) = 2$ condition is not necessary for the following construction, though we would need to modify the system slightly due to the loss of bijectivity in the final map used for the construction of the corresponding public key.

Let $\phi : K \longrightarrow k^n$ be the standard k -linear isomorphism between K and k^n given by

$$\phi(a_0 + a_1x + \cdots + a_{n-1}x^{n-1}) = (a_0, a_1, \dots, a_{n-1}).$$

The subfield k of K is embedded in k^n in the standard way:

$$\phi(a) = (a, 0, \dots, 0), \quad \forall a \in k.$$

Note that here ϕ is a k -linear map if we treat k as a subfield in K .

Choose θ so that $0 < \theta < n$ and

$$\gcd(q^\theta + 1, q^n - 1) = 1,$$

and define the map \tilde{F} over K by

$$\tilde{F}(X) = X^{1+q^\theta}. \tag{2.1}$$

The conditions on θ insure that \tilde{F} is an invertible map; indeed, if t is an integer such that

$$t(1 + q^\theta) \equiv 1 \pmod{q^n - 1},$$

then \tilde{F}^{-1} is simply

$$\tilde{F}^{-1}(X) = X^t.$$

Now let F be the map over k^n defined by

$$F(x_1, \dots, x_n) = \phi \circ \tilde{F} \circ \phi^{-1}(x_1, \dots, x_n) = (f_1, \dots, f_n),$$

where $f_1, \dots, f_n \in k[x_1, \dots, x_n]$. To finish the description of the construction of Matsumoto-Imai, let us now choose L_1 and L_2 to be two invertible affine transformations over k^n . Define the map over k^n by

$$\bar{F}(x_1, \dots, x_n) = L_1 \circ F \circ L_2(x_1, \dots, x_n) = (\bar{f}_1, \dots, \bar{f}_n), \quad (2.2)$$

where $\bar{f}_1, \dots, \bar{f}_n \in k[x_1, \dots, x_n]$. See Figure 2.1 for a commutative diagram that captures the essence of the MI construction.

$$\begin{array}{ccccccc}
 k^n & \xrightarrow{L_2} & k^n & \xrightarrow{\phi^{-1}} & K & \xrightarrow{\bar{F}} & K & \xrightarrow{\phi} & k^n & \xrightarrow{L_1} & k^n \\
 \downarrow id & & id \downarrow & & & & & & id \uparrow & & \uparrow id \\
 & & k^n & \xrightarrow{\quad F \quad} & & & k^n & & & & \\
 k^n & \xrightarrow{\quad \bar{F} \quad} & & & & & & & & & k^n
 \end{array}$$

Figure 2.1. Composition of maps in the construction of MI.

We can now fully describe the Matsumoto-Imai public key cryptosystem.

The Public Key

The public key of MI includes the following:

- 1.) The field k including its additive and multiplicative structure;
- 2.) The n polynomials $\bar{f}_1, \dots, \bar{f}_n \in k[x_1, \dots, x_n]$.

The Private Key

The private key includes the two invertible affine transformations L_1 and L_2 . The parameter θ can be kept private, though this is not critical. Since there are fewer than n choices for θ and n is typically not very large, hiding θ has no substantial effect on attack complexities (only a factor of n).

Encryption

Given a plaintext message (x'_1, \dots, x'_n) , the associated ciphertext is (y'_1, \dots, y'_n) , where

$$y'_i = \bar{f}_i(x'_1, \dots, x'_n),$$

for $i = 1, \dots, n$. This can be done by anyone, since the public key is available to anyone.

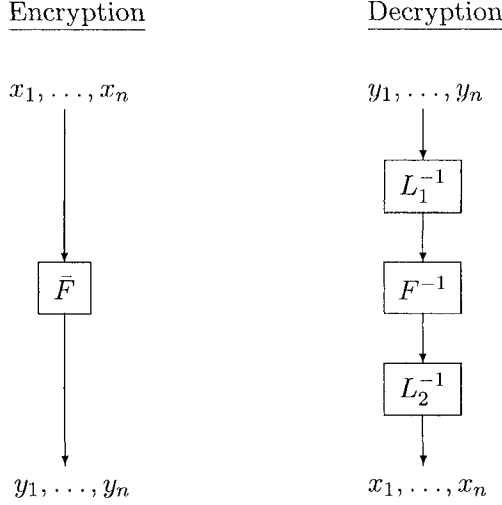


Figure 2.2. Single-branch MI encryption and decryption.

Decryption

We can decrypt the ciphertext (y'_1, \dots, y'_n) by computing

$$\begin{aligned}
 \bar{F}^{-1}(y'_1, \dots, y'_n) &= L_2^{-1} \circ F^{-1} \circ L_1^{-1}(y'_1, \dots, y'_n) \\
 &= L_2^{-1} \circ \phi \circ \tilde{F}^{-1} \circ \phi^{-1} \circ L_1^{-1}(y'_1, \dots, y'_n).
 \end{aligned}$$

In general the components of \bar{F}^{-1} will be of very high degree, and therefore in practice we decrypt the ciphertext (y'_1, \dots, y'_n) by executing the following steps:

- 1.) First compute $(z'_1, \dots, z'_n) = L_1^{-1}(y'_1, \dots, y'_n)$;
- 2.) Then compute $(\bar{z}_1, \dots, \bar{z}_n) = \phi \circ \tilde{F}^{-1} \circ \phi^{-1}(z'_1, \dots, z'_n)$;
- 3.) Finally compute $(x'_1, \dots, x'_n) = L_2^{-1}(\bar{z}_1, \dots, \bar{z}_n)$.

If the corresponding cryptosystem is secure, then this decryption process can be performed only by those who have access to the private key. See Figure 2.2 for a graphical representation of the encryption and decryption process.

Degree of the Public Key Components

The components of the map F are polynomials in $k[x_1, \dots, x_n]$. In fact, since we are thinking of the variables x_1, \dots, x_n as the plaintext

message “bits” in the field k , we will identify f_1, \dots, f_n with the corresponding representative of minimal total degree in the ring of functions from k^n to k

$$\text{Fun}(k^n, k) = k[x_1, \dots, x_n] / (x_1^q - x_1, \dots, x_n^q - x_n),$$

where total degree is defined as usual. For notational convenience, we will abuse notation and use $k[x_1, \dots, x_n]$ instead of $\text{Fun}(k^n, k)$. We shall never use the notation $k[x_1, \dots, x_n]$ for the polynomial ring in the variables x_1, \dots, x_n with coefficients in k unless explicitly announced beforehand. Similarly, the notation $K[X]$ will be used for the ring of functions from K to K ; that is, we identify $K[X]$ with $K[X]/(X^{q^n} - X)$, unless announced otherwise. As such, we shall use the terms “polynomial” and “function” interchangeably. Let us now explore the relationship between the degree of \tilde{F} and the degrees of f_1, \dots, f_n .

The maps $T_i(X) = X^{q^i}$ on K , for $i = 0, 1, \dots, n-1$, are the well-known Frobenius maps. In fact, the set of these maps is exactly the Galois group $G = \text{Gal}(K/k)$, and the group ring $KG = \{\sum_{i=0}^{n-1} \alpha_i T_i \mid \alpha_i \in K\}$ is the set of all k -linear maps on K (see Appendix A). But from this it is easy to see that for any $L(X) \in KG$ we have that $\phi \circ L \circ \phi^{-1}$ is a k -linear map over k^n , hence the components of $\phi \circ L \circ \phi^{-1}$ each have total degree one in $k[x_1, \dots, x_n]$.

In order to better see the relationship between the degree of $H(X) \in K[X]$ and the degree of the components of $\phi \circ H \circ \phi^{-1}$, let us define the q -Hamming weight degree of the monomial $X^e \in K[X]$, where $0 \leq e < q^n$, to be the sum of the coefficients in the base- q expansion of e , also known as the q -Hamming weight of e . The q -Hamming weight degree of a function $H(X) \in K[X]$ is then defined to be the largest q -Hamming weight degree over all monomials of $H(X)$.

Now suppose we have a function $H(X) \in K[X]$ of q -Hamming weight degree d . Then the components of $\phi \circ H \circ \phi^{-1}$ will be of total degree d . In particular, since the q -Hamming weight degree of \tilde{F} is two, it follows that the total degree of each of the f_1, \dots, f_n is two. Since L_1 and L_2 are invertible affine transformations, the total degree of each of the $\tilde{f}_1, \dots, \tilde{f}_n$ is two as well.

A Toy Example

We now illustrate the MI cryptosystem using a toy example with small parameters.

Let $k = GF(2^2)$ be the finite field with $q = 2^2 = 4$ elements. The multiplicative group for the nonzero elements of this field can be generated by the field element α which satisfies $\alpha^2 + \alpha + 1 = 0$. The field elements

+	0	1	α	α^2	*	0	1	α	α^2
0	0	1	α	α^2	0	0	0	0	0
1	1	0	α^2	α	1	0	1	α	α^2
α	α	α^2	0	1	α	0	α	α^2	1
α^2	α^2	α	1	0	α^2	0	α^2	1	α

Table 2.1. Addition and multiplication table of $GF(2^2)$.

of k can be presented as $\{0, 1, \alpha, \alpha^2\}$ and the addition and multiplication tables are given in Table 2.1.

Next choose $n = 3$ and $g(x) = x^3 + x + 1$, an irreducible polynomial in $k[x]$. Set $K = k[x]/(x^3 + x + 1)$. There are only two possible choices for θ ; namely $\theta = 1$ or $\theta = 2$. We will use $\theta = 2$. The map \tilde{F} and its inverse are given by

$$\tilde{F}(X) = X^{1+4^2} \quad \tilde{F}^{-1}(X) = X^{26}.$$

Let L_1 and L_2 be given by

$$L_1(x_1, x_2, x_3) = \begin{pmatrix} \alpha^2 & \alpha & \alpha \\ \alpha & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ \alpha \end{pmatrix}$$

and

$$L_2(x_1, x_2, x_3) = \begin{pmatrix} 1 & 0 & \alpha \\ 0 & 1 & \alpha \\ 1 & \alpha & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} \alpha \\ \alpha^2 \\ \alpha^2 \end{pmatrix}$$

To derive the public key polynomials in terms of the plaintext message variables x_1, x_2, x_3 we begin by computing $\phi^{-1} \circ L_2(x_1, x_2, x_3)$, which we find to be

$$(\alpha + x_1 + \alpha x_3) + (\alpha^2 + x_2 + \alpha x_3)x + (\alpha^2 + x_1 + \alpha x_2)x^2.$$

If we denote this by X , then we next compute $\tilde{F}(X) = X^{1+4^2} = X \cdot X^{16}$. The exponentiation is easily done since we only have to apply it to each term of X . There are no degrees higher than two since we are working in the finite field k of characteristic two. Thus $\tilde{F}(X)$ is

$$\begin{aligned} & 1 + \alpha^2 x_1 + \alpha x_2 + x_3 + x_1 x_2 + \alpha x_1 x_3 + \alpha^2 x_2 x_3 \\ & + (\alpha + \alpha x_1 + x_2 + \alpha^2 x_3 + x_1^2 + \alpha^2 x_1 x_2 + x_2^2 + x_2 x_3)x + (\alpha^2 + \alpha^2 x_1 \\ & + \alpha x_2 + \alpha x_3 + x_1^2 + x_1 x_2 + \alpha x_1 x_3 + \alpha^2 x_2^2 + \alpha x_2 x_3 + \alpha^2 x_3^2)x^2. \end{aligned}$$

Finally we compute $L_1 \circ \phi(X)$ to get the public key polynomials

$$\begin{aligned}\bar{f}_1(x_1, x_2, x_3) &= 1 + x_3 + \alpha x_1 x_3 + \alpha^2 x_2^2 + \alpha^2 x_2 x_3 + x_3^2 \\ \bar{f}_2(x_1, x_2, x_3) &= 1 + \alpha^2 x_1 + \alpha x_2 + x_3 + x_1^2 + x_1 x_2 + \alpha^2 x_1 x_3 + x_2^2 \\ \bar{f}_3(x_1, x_2, x_3) &= \alpha^2 x_3 + x_1^2 + \alpha^2 x_2^2 + x_2 x_3 + \alpha^2 x_3^2,\end{aligned}$$

which will be used to encrypt plaintext messages. If, for example, we wish to encrypt the plaintext $(x'_1, x'_2, x'_3) = (1, \alpha, \alpha^2)$, then we compute

$$\begin{aligned}y'_1 &= \bar{f}_1(1, \alpha, \alpha^2) = 0 \\ y'_2 &= \bar{f}_2(1, \alpha, \alpha^2) = 0 \\ y'_3 &= \bar{f}_3(1, \alpha, \alpha^2) = 1\end{aligned}$$

to get the ciphertext $(0, 0, 1)$.

The person in charge of decrypting this ciphertext knows L_1^{-1} , \tilde{F}^{-1} and L_2^{-1} . With

$$L_1^{-1}(y_1, y_2, y_3) = \begin{pmatrix} \alpha^2 & 1 & 1 \\ 1 & \alpha^2 & \alpha \\ \alpha^2 & 1 & 0 \end{pmatrix} \begin{pmatrix} y_1 - 0 \\ y_2 - 1 \\ y_3 - \alpha \end{pmatrix}$$

and the given ciphertext we first find

$$L_1^{-1}(0, 0, 1) = \begin{pmatrix} \alpha \\ \alpha \\ 1 \end{pmatrix},$$

from which $X = \alpha + \alpha x + x^2$ follows. In this toy example

$$\tilde{F}^{-1}(X) = X^{26} = \alpha + x^2,$$

which can easily be computed by the binary method (also known as the square-and-multiply method). In real applications this approach would be too time consuming, since the exponent t for X is typically very large. Instead one selects a θ where the binary representation of t exhibits a pattern, which then can be exploited to speed up the process of evaluating X^t .

Continuing with the toy example, we now have $(\bar{z}_1, \bar{z}_2, \bar{z}_3) = (\alpha, 0, 1)$. From

$$L_2^{-1}(y_1, y_2, y_3) = \begin{pmatrix} \alpha^2 & \alpha^2 & \alpha \\ \alpha & \alpha & \alpha \\ 1 & \alpha & 1 \end{pmatrix} \begin{pmatrix} y_1 - \alpha \\ y_2 - \alpha^2 \\ y_3 - \alpha^2 \end{pmatrix}.$$

we obtain $L_2^{-1}(\alpha, 0, 1) = (1, \alpha, \alpha^2)^T$, the original plaintext.

Multiple-Branch MI

A multiple-branch cryptosystem is one essentially composed of several basic (single-branch) cryptosystems. The input is partitioned first, with each part sent to its own single branch cipher. The outputs of each branch are then combined into a single output. The input is first transformed, usually in the form of an invertible affine transformation, before being partitioned in order to hide the branches. Similarly, the combination of the outputs from the branches usually undergoes a transformation. See Figure 2.3 for a pictorial illustration of this general idea. Note that if the single-branch ciphers C_1, C_2, \dots, C_b and the input-output transformations are invertible, then the multi-branch cipher will be invertible as well.

In the case of multi-branch MI, each branch will be a basic single-branch MI as described in the previous section. Let b be the number of branches and pick positive integers n_1, \dots, n_b such that $n_1 + \dots + n_b = n$. For each i , pick an irreducible polynomial $g_i(x) \in k[x]$ of degree n_i and define $K_i = k[x]/g_i(x)$. Then K_i is a degree n_i field extension of k , with k -linear isomorphism

$$\phi_i : K_i \longrightarrow k^{n_i}$$

such that

$$\phi_i(a_0 + a_1x + \dots + a_{n_i-1}x^{n_i-1}) = (a_0, a_1, \dots, a_{n_i-1}).$$

As in the case of a single branch, if we choose (independently) the $\theta_1, \dots, \theta_b$ such that $0 < \theta_i < n_i$ and $\gcd(q^{\theta_i} + 1, q^{n_i} - 1) = 1$ for each i , then we can construct the invertible maps

$$\tilde{F}_i(X) = X^{1+q^{\theta_i}}$$

and then

$$F_i = \phi_i \circ \tilde{F}_i \circ \phi_i^{-1} = (f_{i1}, \dots, f_{in_i}),$$

where each f_{1j} is a polynomial in $k[x_1, \dots, x_{n_1}]$, for $j = 1, \dots, n_1$; each f_{2j} is a polynomial in $k[x_{n_1+1}, \dots, x_{n_1+n_2}]$, for $j = 1, \dots, n_2$; ...; and each f_{bj} is a polynomial in $k[x_{n-n_b+1}, \dots, x_n]$ for $j = 1, \dots, n_b$.

We then combine the branches together to define a new map F over k^n by

$$\begin{aligned} F(x_1, \dots, x_n) &= (F_1, F_2, \dots, F_b) \\ &= (f_{11}, \dots, f_{1n_1}, f_{21}, \dots, f_{2n_2}, \dots, f_{b1}, \dots, f_{bn_b}), \end{aligned} \quad (2.3)$$

and choose L_1 and L_2 to be invertible affine transformations on k^n . Finally define the map \bar{F} over k^n as before:

$$\bar{F}(x_1, \dots, x_n) = L_1 \circ F \circ L_2(x_1, \dots, x_n) = (\bar{f}_1, \dots, \bar{f}_n),$$

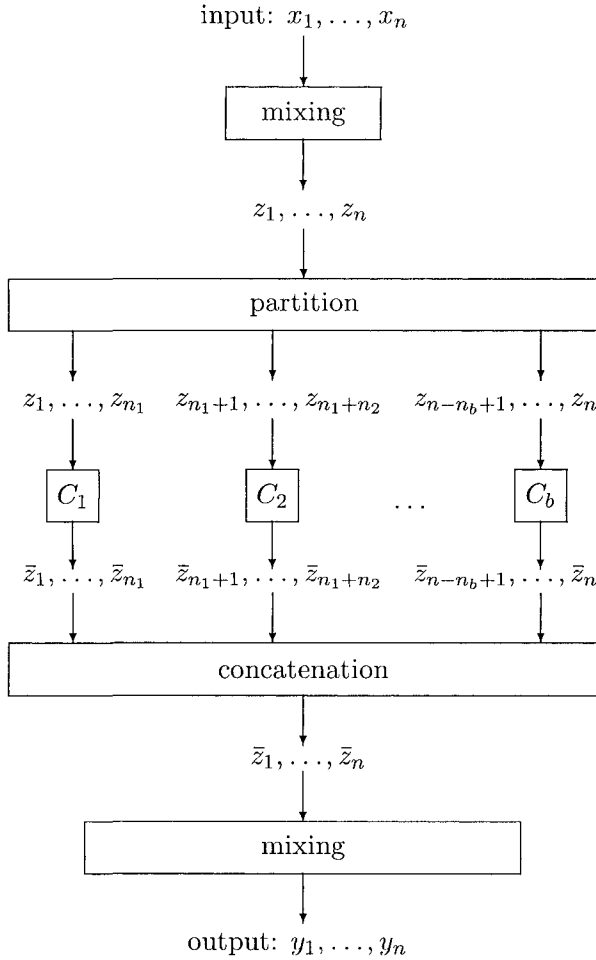


Figure 2.3. A multi-branch cipher composed of single-branch ciphers C_1, C_2, \dots, C_b .

where each \bar{f}_i is a degree two polynomial in $k[x_1, \dots, x_n]$.

We can see that a multiple-branch implementation of MI is essentially the image of several single-branch MI implementations under an invertible affine transformation. Though it may seem that multiple branches provide more security, we shall see later that this is not the case.

2.2 Key Size and Efficiency of MI

The public key of the Matsumoto-Imai cryptosystem is a set of degree two polynomials $\bar{f}_1, \dots, \bar{f}_n \in k[x_1, \dots, x_n]$. Each polynomial has $1 + n +$

$n(n+1)/2 = (n+1)(n+2)/2$ terms, hence the public key amounts to a set of $n(n+1)(n+2)/2$ coefficients in k when $q > 2$. For $q = 2$ the key size will be smaller because there are no square terms due to the fact that $x_i^2 = x_i$.

This is rather large compared with that of RSA, even if we choose k to be $GF(2^8)$ and $n = 32$, the parameters originally suggested by Matsumoto-Imai in 1988. However, with systems like RSA there are other considerations, in particular the implementation software, whereas with MPKCs the implementation requires minimum work beyond the public key.

Though the public key of MI may be large compared with other schemes such as RSA, the great advantage of MI lies in its computational efficiency. If we choose $q = |k|$ to be small, then we can store the multiplication table in memory using the fact that the nonzero elements of k form a cyclic multiplicative group. This makes the encryption much faster than schemes like RSA which must work with large integers. This technical detail can also be used in the decryption process, including the most expensive calculation in computing with \bar{F}^{-1} . In fact, MI originally generated a lot of excitement precisely because the practical implementations first suggested were far faster than RSA and promised the same level of security.

The Matsumoto-Imai cryptosystem was proposed in 1988 [Matsumoto and Imai, 1988], and was considered as one of the candidates for the Japanese government security standard. However, MI was defeated in 1995 by Patarin's algebraic attack via linearization equations [Patarin, 1995].

2.3 Linearization Equations Attack

We begin by defining the notion of a linearization equation (LE) in a general way.

Definition 2.3.1. *Let $\mathcal{G} = \{g_1, \dots, g_m\}$ be any set of m polynomials in $k[x_1, \dots, x_n]$. A linearization equation for \mathcal{G} is any polynomial in $k[x_1, \dots, x_n, y_1, \dots, y_m]$ of the form*

$$\sum_{i=1}^n \sum_{j=1}^m a_{ij} x_i y_j + \sum_{i=1}^n b_i x_i + \sum_{j=1}^m c_j y_j + d, \quad (2.4)$$

such that we obtain the zero function in $k[x_1, \dots, x_n]$ upon substituting in g_j for y_j , for $j = 1, \dots, m$. Equivalently, a linearization equation

is any equation in $k[x_1, \dots, x_n]$ of the form

$$\sum_{i=1}^n \sum_{j=1}^m a_{ij} x_i g_j(x_1, \dots, x_n) + \sum_{i=1}^n b_i x_i + \sum_{j=1}^m c_j g_j(x_1, \dots, x_n) + d = 0$$

which holds for all $(x'_1, \dots, x'_n) \in k^n$.

It is clear that for a given \mathcal{G} , the set of all linearization equations of \mathcal{G} forms a k -vector space. This space will be referred to as the linearization equation space of \mathcal{G} .

Patarin keenly observed that the linearization equation space for the components of \bar{F} can be used to attack the Matsumoto-Imai cryptosystems. To see this, let $\{\bar{f}_1, \dots, \bar{f}_n\}$ be the set of components of \bar{F} , and suppose we have a linearization equation of this set of the form of (2.4). For a given ciphertext (y'_1, \dots, y'_n) , substituting in y'_i for \bar{f}_i produces a linear (hopefully nontrivial) equation in the variables x_1, \dots, x_n whose solution set contains the plaintext.

With enough linearization equations, we can hope to produce enough linear equations such that the resulting system has the desired plaintext as its unique solution. Even if we cannot find directly the plaintext from these linear equations for a given ciphertext, as long as the LEs can produce enough linearly independent linear equations for the corresponding plaintext, these linear equations can then be plugged into the quadratic public equations derived from the public key and the ciphertext to reduce the number of variables and make it much easier to solve it. To decide the feasibility of this attack, we must first find the number of linearly independent linear equations we can hope to derive from the space of linearization equations of the components of \bar{F} . We begin the analysis by considering the single-branch case of MI.

Linearization Equations of Single-Branch MI

The following theorem gives a lower bound on the number of linearly independent linear equations that we can generate from the components of \bar{F} .

Theorem 2.3.1. *Let $\{\bar{f}_1, \dots, \bar{f}_n\}$ be the public key for a single-branch implementation of MI. Fix a ciphertext $Y' = (y'_1, \dots, y'_n) \in k^n$ and let $\bar{\mathcal{L}}$ be the space of linearization equations of $\{\bar{f}_1, \dots, \bar{f}_n\}$. If $\bar{\mathcal{L}}_{Y'}$ is the space of equations that are derived by substituting in y'_i for y_i (for $i = 1, \dots, n$) in each equation of $\bar{\mathcal{L}}$, then the number of linearly independent linear equations in $\bar{\mathcal{L}}_{Y'}$ is at least*

$$n - \gcd(n, \theta) \geq \frac{2n}{3}.$$

The exceptional case is $L^{-1}(Y') = (0, \dots, 0)$ when there are only trivial equations.

To prove this theorem we will need the following two lemmas.

Lemma 2.3.1. *Let $\bar{F} = L_1 \circ F \circ L_2$ be as in the construction of single-branch MI. Let \mathcal{L} be the space of linearization equations of $\{f_1, \dots, f_n\}$ and let $\bar{\mathcal{L}}$ be the space of linearization equations of $\{\bar{f}_1, \dots, \bar{f}_n\}$. Then these two k -vector spaces have the same dimension; i.e.,*

$$\dim_k \mathcal{L} = \dim_k \bar{\mathcal{L}}.$$

Proof. First suppose that L_2 is the identity, so that

$$\bar{f}_j(x_1, \dots, x_n) = \sum_{i=1}^n \alpha_{ij} f_i(x_1, \dots, x_n) + \beta_j.$$

Then

$$\begin{aligned} 0 &= \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i \bar{f}_j + \sum_{i=1}^n b_i x_i + \sum_{j=1}^n c_j \bar{f}_j + d \\ &= \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i \left(\sum_{l=1}^n \alpha_{jl} f_l + \beta_j \right) + \sum_{i=1}^n b_i x_i + \sum_{j=1}^n c_j \left(\sum_{l=1}^n \alpha_{jl} f_l + \beta_j \right) \\ &\quad + d \\ &= \sum_{i=1}^n \sum_{j=1}^n a'_{ij} x_i f_j + \sum_{i=1}^n b'_i x_i + \sum_{j=1}^n c'_j f_j + d', \end{aligned}$$

a linearization equation for f_1, \dots, f_n .

Similarly, by looking at $F = L_1^{-1} \circ \bar{F}$ and starting with a linearization equation for f_1, \dots, f_n , we can derive a linearization equation for $\bar{f}_1, \dots, \bar{f}_n$. From this bijection we see that the dimension of the linearization equations for F and $L_1 \circ F$ are the same.

Now suppose that L_1 is the identity, and let

$$\bar{x}_j = \sum_{i=1}^n \alpha_{ij} x_i + \beta_j,$$

so that

$$\bar{f}_i(x_1, \dots, x_n) = f_i(\bar{x}_1, \dots, \bar{x}_n).$$

Then

$$0 = \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i f_j(x_1, \dots, x_n) + \sum_{i=1}^n b_i x_i + \sum_{j=1}^n c_j f_j(x_1, \dots, x_n) + d$$

which gives

$$0 = \sum_{i=1}^n \sum_{j=1}^n a_{ij} \bar{x}_i f_j(\bar{x}_1, \dots, \bar{x}_n) + \sum_{i=1}^n b_i \bar{x}_i + \sum_{j=1}^n c_j f_j(\bar{x}_1, \dots, \bar{x}_n) + d,$$

since the invertible change of variables amounts to a permutation on k^n . But then we have

$$0 = \sum_{i=1}^n \sum_{j=1}^n a_{ij} \bar{x}_i \bar{f}_j(x_1, \dots, x_n) + \sum_{i=1}^n b_i \bar{x}_i + \sum_{j=1}^n c_j \bar{f}_j(x_1, \dots, x_n) + d,$$

which, as above, can be rewritten as

$$0 = \sum_{i=1}^n \sum_{j=1}^n a'_{ij} x_i \bar{f}_j + \sum_{i=1}^n b'_i x_i + \sum_{j=1}^n c'_j \bar{f}_j + d',$$

a linearization equation for $\bar{f}_1, \dots, \bar{f}_n$.

Similarly, by looking at $F = \bar{F} \circ L_2^{-1}$ and starting with a linearization equation for $\bar{f}_1, \dots, \bar{f}_n$, we can derive a linearization equation for f_1, \dots, f_n . From this bijection we see that the dimension of the linearization equations for F and $F \circ L_2$ are the same.

Finally, we conclude that $\dim_k \mathcal{L} = \dim_k \bar{\mathcal{L}}$. \square

Lemma 2.3.2. *Let \mathcal{L} and $\bar{\mathcal{L}}$ be as in the previous lemma, fix a ciphertext $Y' = (y'_1, \dots, y'_n) \in k^n$, and let $Z = L_1^{-1}(Y') = (z_1, \dots, z_n)$. Let \mathcal{L}_Z be the space of linear equations that arise from substituting in z_i for y_i (for $i = 1, \dots, n$) in each linearization equation in \mathcal{L} , and let $\bar{\mathcal{L}}_{Y'}$ be the space of linear equations that arise from substituting in y'_i for y_i (for $i = 1, \dots, n$) in each linearization equation in $\bar{\mathcal{L}}$. Then these two k -vector spaces have the same dimension; i.e.,*

$$\dim_k \mathcal{L}_Z = \dim_k \bar{\mathcal{L}}_{Y'}.$$

Proof. In the proof of the previous lemma we constructed a bijection between \mathcal{L} and $\bar{\mathcal{L}}$. This induces a bijection between \mathcal{L}_Z and $\bar{\mathcal{L}}_{Y'}$ from which the result follows. \square

To see how Patarin first constructed linearization equations, we let $X, Y \in K$ be such that

$$Y = \tilde{F}(X) = X^{q^\theta + 1}.$$

We then have

$$\begin{aligned} Y^{q^\theta - 1} &= (X^{q^\theta + 1})^{q^\theta - 1} \\ &= X^{(q^\theta + 1)(q^\theta - 1)} \\ &= X^{q^{2\theta} - 1}. \end{aligned}$$

If we multiply both sides by XY , we see that

$$XY^{q^\theta} = X^{q^{2\theta}}Y,$$

or equivalently,

$$XY^{q^\theta} - X^{q^{2\theta}}Y = 0.$$

Finally define $\tilde{R}(X, Y) \in K[X, Y]$ by

$$\tilde{R}(X, Y) = XY^{q^\theta} - X^{q^{2\theta}}Y,$$

and

$$R = \phi \circ \tilde{R} \circ (\phi^{-1} \times \phi^{-1}) \quad (2.5)$$

From this R we can derive n linearization equations for the components of F . Specifically, the n components of $R(x_1, \dots, x_n, y_1, \dots, y_n)$ are of the form (2.4), and, by construction, substituting in f_i for y_i (for $i = 1, \dots, n$) yields the zero polynomial in $k[x_1, \dots, x_n]$.

It is natural to ask how many linearly independent linear equations arise from R for a specific $(y'_1, \dots, y'_n) \in k^n$. Let $(x'_1, \dots, x'_n) \in k^n$ be $F^{-1}(y'_1, \dots, y'_n)$, and let $Y' = \phi^{-1}(y'_1, \dots, y'_n)$ and $X' = \phi^{-1}(x'_1, \dots, x'_n)$. Then X' must be a solution of

$$X^{q^{2\theta}}Y' = X(Y')^{q^\theta}, \quad (2.6)$$

or

$$X^{q^{2\theta}-1} = (Y')^{q^\theta-1}, \quad (2.7)$$

if $Y' \neq 0$. But the second equation has at most $\gcd(q^{2\theta} - 1, q^n - 1)$ solutions in K . Furthermore, because of the condition $\gcd(q^\theta + 1, q^n - 1) = 1$, we have that

$$\gcd(q^{2\theta} - 1, q^n - 1) = \gcd(q^\theta - 1, q^n - 1),$$

hence (2.6) has at most $\gcd(q^\theta - 1, q^n - 1) + 1$ solutions, including the trivial solution. To find this number explicitly we will need the following lemma, which is easily proved.

Lemma 2.3.3. *For any two positive integers a, b we have*

$$\gcd(q^a - 1, q^b - 1) = q^{\gcd(a, b)} - 1.$$

In particular, the lemma tells us that the total number of solutions for (2.6) is at most $q^{\gcd(\theta, n)}$. If λ is the number of linearly independent linear equations that arise from (2.6), then there will be $q^{n-\lambda}$ solutions to the

corresponding system of linear equations. Therefore $q^{n-\lambda} \leq q^{\gcd(\theta, n)}$, and so $\lambda \geq n - \gcd(\theta, n)$.

The three largest possible values of $\gcd(\theta, n)$ are n , $n/2$ if n is even, and $n/3$ if 3 divides n , and the rest are of all smaller values. Therefore, if we show that the first two cases are impossible, then we can conclude that

$$n - \gcd(\theta, n) \geq \frac{2n}{3}.$$

First we know that it is impossible that $\gcd(\theta, n)$ is n , because of the choice of θ is larger than 0 and less than n . Second, if $\gcd(\theta, n) = n/2$, this means that θ must be $n/2$ itself. Then we know that

$$\gcd(q^{n/2} + 1, q^n - 1) = q^{n/2} + 1 > 1,$$

which contradicts the invertibility condition which requires that

$$\gcd(q^\theta + 1, q^n - 1) = 1.$$

Therefore $\gcd(\theta, n)$ cannot be $n/2$ either and the largest possible value for $\gcd(\theta, n)$ is $n/3$.

This proves the following theorem, which combined with Lemma 2.3.2 gives us a proof of Theorem 2.3.1. The exceptional case in Theorem 2.3.1 is $L_1^{-1}(Y') = (0, \dots, 0)$ and all linear equations derived from the linearization equation are again trivial ones, $0 = 0$.

Theorem 2.3.2. *Let \mathcal{L} be the space of linearization equations for the components of F and fix $Y' = (y'_1, \dots, y'_n) \in k^n$. If $\mathcal{L}_{Y'}$ is the space of linear equations resulting from substituting in y'_i for y_i (for $i = 1, \dots, n$) in each element of \mathcal{L} , then $\dim_k \mathcal{L}_{Y'}$ is at least*

$$n - \gcd(\theta, n) \geq \frac{2n}{3},$$

except when $Y' = (0, \dots, 0)$.

If $\gcd(\theta, n) = 1$ then it is clear that we can defeat the system easily using linearization equations alone. More generally, we see that the single branch Matsumoto-Imai cryptosystem is not very secure since for a given ciphertext we can always find at least $2n/3$ linear equations satisfied by the plaintext, which is analogous to leaking $2/3$ of the information. More importantly, these equations can be used to eliminate $2/3$ of the variables of the quadratic public equations derived from the public key and the ciphertext, which should then be much easier to solve than before.

The next question we consider is how to actually generate linearization equations. We explain two different approaches: one based on plaintext-ciphertext pairs, and the other based on the structure of polynomial functions.

Plaintext-Ciphertext Pairs

Using the public key we can generate several plaintext-ciphertext pairs. For each pair given by $\bar{F}(x'_1, \dots, x'_n) = (y'_1, \dots, y'_n)$, we can substitute in x'_i for x_i and y'_j for y_j into the generic linearization equation

$$\sum a_{ij}x_iy_j + \sum b_ix_i + \sum c_jy_j + d = 0,$$

to get a linear equation in the $(n+1)^2$ unknowns $a_{ij}, b_i, c_j, d \in k$. Therefore, if we choose roughly $(n+1)^2$ plaintext-ciphertext pairs, then it is very likely that we can solve the resulting system for the unknown coefficients. The total cost of this process includes:

- 1.) Computation of $(n+1)^2$ plaintext-ciphertext pairs, which has complexity $O(n^4)$;
- 2.) Solving a set $(n+1)^2$ linear equations in $(n+1)^2$ variables, which has complexity $O(n^6)$.

This can be done relatively easily.

Structure of Polynomial Functions

We begin with a generic linearization equation for the components of \bar{F} :

$$\sum a_{ij}x_i\bar{f}_j + \sum b_ix_i + \sum c_j\bar{f}_j + d = 0.$$

As before, we treat the coefficients a_{ij}, b_i, c_j, d as variables taking values in k . After rewriting the left-hand side of this equation as a sum of monomials in the variables x_1, \dots, x_n , we have an equation of the form:

$$\sum \alpha_{ijl}x_ix_jx_l + \sum \beta_{ij}x_ix_j + \sum \gamma_ix_i + \delta = 0, \quad (2.8)$$

where the coefficients $\alpha_{ijl}, \beta_{ij}, \gamma_i, \delta$ are linear functions in the unknown coefficients a_{ij}, b_i, c_j, d .

Remark 2.3.1. *If $q = 2$, then we should make use of the fact that $x^3 = x^2 = x$ for any $x \in k$. In particular, any power of x_i occurring in (2.8) will be replaced by x_i , for $i = 1, \dots, n$.*

From the theory of polynomials over a finite field, we know that each of the $\alpha_{ijk}, \beta_{ij}, \gamma_i, \delta$ must be equal to zero, which produces $\frac{(n+1)(n+2)(n+3)}{6}$

linear equations in the unknown coefficients a_{ij}, b_i, c_j, d , when $q > 2$. The solution set for this system of equations is then used to construct linearization equations.

It is very likely that we will not need to use all $(n+1)(n+2)(n+3)/6$ linear equations, and that we probably only need roughly $(n+1)^2$ of them. We can also confirm easily if indeed we have the right solution space, if we know the dimension of the space of linearization equations (we will say more in the next subsection about how to calculate this dimension). If the dimension of the space is too large, we can always add more equations until the right solution space is found.

Here the main cost is to solve a set of $(n+1)^2$ linear equations in $(n+1)^2$ variables. As before, the complexity of this is $O(n^6)$.

Dimension of the Space of Linearization Equations for Basic MI

Now we will present the results related to calculation of the dimension of the space of linearization equations as presented in [Diene et al., 2006].

Theorem 2.3.3. *Let \mathcal{L} be the space of linearization equations associated with the components of a given invertible Matsumoto-Imai map \bar{F} (hence we may assume that $\theta \neq n/2$). If $q > 2$, then*

$$\dim_k \mathcal{L} = \begin{cases} 2n/3, & \text{if } \theta = n/3, 2n/3; \\ n, & \text{otherwise.} \end{cases}$$

If $q = 2$ and $\theta = n/3, 2n/3$, then

$$\dim_k \mathcal{L} = \begin{cases} 7, & \text{if } n = 6, \theta = 2, 4; \\ 8, & \text{if } n = 3, \theta = 1, 2; \\ 2n/3, & \text{otherwise.} \end{cases}$$

If $q = 2$ and $\theta \neq n/3, 2n/3$, then

$$\dim_k \mathcal{L} = \begin{cases} 10, & \text{if } n = 4, \theta = 1, 3; \\ 2n, & \text{if } \theta = 1, n-1, (n \pm 1)/2; \\ 3n/2, & \text{if } \theta = (n \pm 2)/2; \\ n, & \text{otherwise.} \end{cases}$$

The key idea used in the calculation of $\dim_k \mathcal{L}$ is to lift the problem to an extension field. The approach is very similar to that used by Kipnis and Shamir in [Kipnis and Shamir, 1999]. We present only a sketch of the proof of the case where $q > 2$; see [Diene et al., 2006] for the complete proof of Theorem 2.3.3.

The proof in [Diene et al., 2006] uses some very abstract mathematical concepts and theorems, which look simple but may be difficult for people who are not very familiar with the related mathematical theory. Our proof here is more direct and more from the point of computation.

Recall $\tilde{R} : K \times K \longrightarrow K$ is defined by

$$\tilde{R}(X, Y) = XY^{q^\theta} - X^{q^{2\theta}}Y,$$

and $R : k^{2n} \longrightarrow k^n$ is defined by

$$R = \phi \circ \tilde{R} \circ (\phi^{-1} \times \phi^{-1}) = (r_1, \dots, r_n),$$

where $r_1, \dots, r_n \in k[x_1, \dots, x_n, y_1, \dots, y_n]$. The first step is to show that the n linearization equations derived from R are linearly independent if $q > 2$ and $\theta \neq n/3, 2n/3$. We will show this by way of contradiction, so let us assume that these n linearization equations are not linearly independent. In this case there must exist a nonzero vector $(\alpha_1, \dots, \alpha_n) \in k^n$ such that $\alpha_1 r_1 + \dots + \alpha_n r_n = 0$ in the polynomial ring $k[x_1, \dots, x_n, y_1, \dots, y_n]$.

Let $L : k^n \longrightarrow k^n$ be the linear map defined by

$$L(x_1, \dots, x_n) = (\alpha_1 x_1 + \dots + \alpha_n x_n, 0, \dots, 0),$$

hence $L \circ R$ is the zero function from k^{2n} to k^n . From this it follows that $\phi^{-1} \circ L \circ \phi \circ \tilde{R}$ is the zero function from $K \times K$ to K since

$$\begin{aligned} \phi^{-1} \circ L \circ \phi \circ \tilde{R} &= (\phi^{-1} \circ L \circ \phi) \circ (\phi^{-1} \circ R \circ (\phi \times \phi)) \\ &= \phi^{-1} \circ (L \circ R) \circ (\phi \times \phi) \\ &= \phi^{-1} \circ 0 \circ (\phi \times \phi). \end{aligned}$$

Now from Lemma A.0.1 and its corollary, there exists a nonzero vector in K^n , say (A_0, \dots, A_{n-1}) , such that

$$\phi^{-1} \circ L \circ \phi(X) = \sum_{i=0}^{n-1} A_i X^{q^i},$$

hence

$$\sum_{i=0}^{n-1} A_i (XY^{q^\theta} - X^{q^{2\theta}}Y)^{q^i} = 0.$$

It is not hard to see that if $q > 2$ and $i \neq 0$ then

$$XY^{q^\theta} \neq (YX^{q^{2\theta}})^{q^i} = Y^{q^i} X^{q^{2\theta+i}},$$

unless $3\theta = n, 2n$. Since we have assumed otherwise, the monomials in this polynomial are linearly independent, and hence all A_i are zero. This contradicts our assumption, and thus the n linearization equations are linearly independent.

To prove that there are no other linearization equations is very similar. Pick any linearization equation, say

$$\sum_{i=1}^n a_{ij}x_iy_j + \sum_{i=1}^n b_ix_i + \sum_{i=1}^n c_jy_j + d = 0$$

so that

$$\sum_{i=1}^n a_{ij}x_i\bar{f}_j + \sum_{i=1}^n b_ix_i + \sum_{i=1}^n c_j\bar{f}_j + d = 0$$

in $k[x_1, \dots, x_n]$, and not all the $a_{ij}, b_i, c_j, d \in k$ are zero.

The map Q taking $(x_1, \dots, x_n, y_1, \dots, y_n)$ to

$$\left(\sum a_{ij}x_iy_j + \sum b_ix_i + \sum c_jy_j + d, 0, \dots, 0 \right) \quad (2.9)$$

is a nonzero map from k^{2n} to k^n . Hence by Lemma A.0.3 in Appendix A, there exists a corresponding unique map \bar{Q} from $K \times K$ to K :

$$\bar{Q} = \phi \circ Q \circ (\phi^{-1} \times \phi^{-1})$$

such that

$$\bar{Q}(X, Y) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} A_{ij} X^{q^i} Y^{q^j} + \sum_{i=0}^{n-1} B_i X^{q^i} + \sum_{j=0}^{n-1} C_j Y^{q^j} + D,$$

where not all the $A_{ij}, B_i, C_j, D \in K$ are zero, and $X = \phi^{-1}(x_1, \dots, x_n)$ and $Y = \phi^{-1}(y_1, \dots, y_n)$.

Because \bar{Q} is derived from a linearization equation, when we substitute in Y for $X^{q^\theta+1}$ in this expression, then we will have the zero function from $K \times K$ to K . Via a direct computation we can show that it will be in the form

$$\sum_{i=0}^{n-1} A_i (XY^{q^\theta} - X^{q^{2\theta}}Y)^{q^i} = 0,$$

if $q > 2$ and $\theta \neq n/3, 2n/3$. From this we conclude that all linearization equations for \bar{F} are linear combinations of the n components of R , and that the dimension of the space of linearization equations is n in the case of $q > 2$ and $\theta \neq n/3, 2n/3$.

Linearization Equations Toy Example

We will illustrate the use of the linearization equations with a small example. We will again use the field $GF(2^2)$ for k , whose field operations are given in Table 2.1. The plaintext is given by $n = 5$ variables $(x_1, x_2, x_3, x_4, x_5) \in k^5$. In order to represent the public key in a more compact form we introduce the additional value $x_0 = 1$, so that the public key can be written as a sum of quadratic terms. With the row vector $\mathbf{x} = (x_0, x_1, x_2, x_3, x_4, x_5)$ the public key is given by

$$y_1 = \mathbf{x} \begin{pmatrix} 0 & 0 & \alpha & 1 & 1 & 1 \\ & \alpha & \alpha & \alpha^2 & \alpha & 0 \\ & & 1 & \alpha^2 & 0 & \alpha \\ & & & \alpha^2 & \alpha & \alpha \\ & & & & \alpha & \alpha^2 \\ & & & & & 1 \end{pmatrix} \mathbf{x}^T, \quad (2.10)$$

$$y_2 = \mathbf{x} \begin{pmatrix} \alpha & 0 & 0 & 0 & \alpha^2 & 1 \\ & \alpha & 0 & \alpha^2 & \alpha^2 & 1 \\ & & 1 & \alpha^2 & 0 & 0 \\ & & & 0 & \alpha & 0 \\ & & & & 1 & \alpha^2 \\ & & & & & 1 \end{pmatrix} \mathbf{x}^T, \quad (2.11)$$

$$y_3 = \mathbf{x} \begin{pmatrix} 1 & \alpha^2 & \alpha & \alpha^2 & 1 & \alpha^2 \\ & \alpha^2 & 0 & 0 & 1 & \alpha^2 \\ & & \alpha^2 & 0 & 1 & 1 \\ & & & \alpha^2 & \alpha & \alpha \\ & & & & 0 & \alpha^2 \\ & & & & & \alpha^2 \end{pmatrix} \mathbf{x}^T, \quad (2.12)$$

$$y_4 = \mathbf{x} \begin{pmatrix} 1 & \alpha^2 & 1 & \alpha^2 & 0 & 0 \\ & \alpha^2 & \alpha & \alpha^2 & \alpha^2 & \alpha^2 \\ & & 1 & 0 & \alpha^2 & \alpha \\ & & & 1 & \alpha^2 & \alpha \\ & & & & 0 & \alpha^2 \\ & & & & & 1 \end{pmatrix} \mathbf{x}^T, \quad (2.13)$$

$$y_5 = \mathbf{x} \begin{pmatrix} \alpha^2 & \alpha^2 & \alpha & 1 & 1 & \alpha^2 \\ & 0 & 0 & \alpha^2 & \alpha^2 & 0 \\ & & \alpha^2 & \alpha & 1 & 1 \\ & & & 1 & 0 & \alpha^2 \\ & & & & \alpha & \alpha \\ & & & & & \alpha^2 \end{pmatrix} \mathbf{x}^T. \quad (2.14)$$

The entries left blank in the matrices are zero, and they will not be stored in a real life application. Assume that a plain text produced the cipher text $(1, 0, 0, 0, 1)$. We will show how to recover the plain text with the help of the linearization equations.

Introduce the value $y_0 = 1$ so that the public key can be represented by the row vector $y = (y_0, y_1, y_2, y_3, y_4, y_5)$. The linearization equations (2.4), which in our case use $m = 5$ and $n = 5$, can now be written in matrix form

$$xAy^T = 0 \quad (2.15)$$

where A is a 6×6 matrix with unknown coefficients $A_{i,j}$, $i, j = 0, \dots, 5$. For setting up the system of linear equations it is easier if the $(m+1)(n+1)$ unknowns are represented by a one dimensional array. With a notation commonly used in programming we introduce the correspondence

$$A_{i,j} \iff A[(m+1)i + j] = 0$$

so that we have the following correspondence for the unknowns appearing in (2.4)

$$\begin{aligned} a_{ij} &\iff A[(m+1)i + j] && \text{for } i = 1, \dots, n; j = 1, \dots, m; \\ b_i &\iff A[(m+1)i] && \text{for } i = 1, \dots, n; \\ c_j &\iff A[j] && \text{for } j = 1, \dots, m; \\ d &\iff A[0]. \end{aligned}$$

Substituting the public key into (2.15) produces a homogeneous polynomial, which is cubic in x_i for $i = 0, \dots, 5$. Collecting the coefficients of the 56 different terms, we obtain a homogeneous system of linear equations in the 36 unknowns $A[0]$ to $A[35]$. The rank of the corresponding matrix is 31, so that the dimension of the linearization equations is $36 - 31 = 5$, which is the common case as predicted by Theorem 2.3.3.

Reducing the matrix to row echelon form we obtain the following

$$\begin{aligned} A[0] &= \alpha A[29] + \alpha^2 A[32] + A[34] + A[35] \\ A[1] &= A[29] + \alpha^2 A[32] + A[33] + A[34] + A[35] \\ A[2] &= \alpha A[29] + A[32] + A[35] \\ A[3] &= A[29] + \alpha A[32] + \alpha^2 A[33] + \alpha^2 A[34] + \alpha A[35] \\ A[4] &= \alpha A[32] + \alpha^2 A[33] + \alpha A[34] + A[35] \\ A[5] &= \alpha A[32] \\ A[6] &= A[29] + A[32] + \alpha^2 A[33] + A[34] + A[35] \\ A[7] &= A[32] + A[33] + \alpha^2 A[34] \\ A[8] &= \alpha^2 A[32] + \alpha A[35] \end{aligned}$$

$$\begin{aligned}
A[9] &= A[29] + A[32] + \alpha^2 A[33] + A[34] + A[35] \\
A[10] &= A[29] + \alpha^2 A[34] \\
A[11] &= \alpha^2 A[29] + \alpha A[35] \\
A[12] &= \alpha^2 A[34] + \alpha^2 A[35] \\
A[13] &= A[29] + A[32] + A[33] + \alpha^2 A[34] \\
A[14] &= \alpha A[32] + \alpha^2 A[35] \\
A[15] &= \alpha^2 A[29] + \alpha^2 A[32] + \alpha^2 A[33] + \alpha^2 A[34] + \alpha^2 A[35] \\
A[16] &= \alpha^2 A[29] + \alpha A[32] + \alpha^2 A[33] + A[34] + A[35] \\
A[17] &= \alpha A[32] \\
A[18] &= \alpha^2 A[29] + \alpha^2 A[32] + \alpha^2 A[33] \\
A[19] &= A[32] + A[35] \\
A[20] &= A[29] + A[32] + \alpha^2 A[33] + \alpha^2 A[34] \\
A[21] &= A[29] + A[32] + \alpha A[33] \\
A[22] &= \alpha^2 A[32] + A[33] + \alpha A[34] + \alpha^2 A[35] \\
A[23] &= \alpha^2 A[29] + \alpha^2 A[32] + \alpha A[34] \\
A[24] &= A[32] + A[34] + A[35] \\
A[25] &= \alpha^2 A[32] + \alpha^2 A[35] \\
A[26] &= \alpha A[29] + A[32] + A[35] \\
A[27] &= \alpha^2 A[32] + \alpha A[33] + \alpha A[34] + \alpha^2 A[35] \\
A[28] &= A[29] + A[34] \\
A[30] &= \alpha A[33] + \alpha A[34] + \alpha A[35] \\
A[31] &= A[32] + \alpha^2 A[33] + A[34] + \alpha^2 A[35]
\end{aligned}$$

where $A[29]$, $A[32]$, $A[33]$, $A[34]$ and $A[35]$ are free parameters. These values and the given cipher text

$$y = (1, y'_1, y'_2, y'_3, y'_4, y'_5) = (1, 1, 0, 0, 0, 1)$$

are now substituted back into (2.15), and the coefficients of the free parameters $A[29]$, $A[32]$, $A[33]$, $A[34]$, $A[35]$ are set to zero to give the following set of equations for the plaintext:

$$\begin{aligned}
\alpha x_1 + x_2 + x_4 + \alpha^2 &= 0, \\
\alpha^2 x_2 + x_3 + \alpha x_4 + x_5 + \alpha &= 0, \\
\alpha x_1 + x_2 + \alpha^2 x_3 + x_5 + 1 &= 0, \\
\alpha x_1 + \alpha x_3 + x_4 + \alpha^2 x_5 &= 0, \\
\alpha^2 x_1 + \alpha^2 x_2 + x_3 + \alpha x_4 &= 0.
\end{aligned}$$

The system of equations has the following solution

$$x_1 = \alpha x_5 + \alpha^2, \quad (2.16)$$

$$x_2 = x_5 + \alpha, \quad (2.17)$$

$$x_3 = x_5 + \alpha^2, \quad (2.18)$$

$$x_4 = \alpha x_5. \quad (2.19)$$

Finally we can find the value of the plaintext in one of two ways.

In the first method we try all possible values of $x_5 \in k$ in order to find out which of the possible plaintexts produced the given ciphertext. With the different values for x_5 in the solutions (2.16) to (2.19) and the public key in (2.10) to (2.14) we find the following possibilities:

plaintext		ciphertext
$(\alpha^2, \alpha, \alpha^2, 0, 0)$	\implies	$(1, 0, 0, 0, 1)$
$(1, \alpha^2, \alpha, \alpha, 1)$	\implies	$(0, \alpha, 0, \alpha^2, \alpha)$
$(0, 0, 1, \alpha^2, \alpha)$	\implies	$(\alpha, 1, 0, \alpha, \alpha^2)$
$(\alpha, 1, 0, 1, \alpha^2)$	\implies	$(\alpha^2, \alpha^2, 0, 1, 0)$

Only the first case produces the given ciphertext and thus we know that the original plaintext was $(\alpha^2, \alpha, \alpha^2, 0, 0)$.

In the other method we substitute the linear equations (2.16) to (2.19) into the public key (2.10) to (2.14) and set it equal to the given ciphertext, that is

$$\begin{aligned} y_1 &= 1, \\ y_2 &= 0, \\ y_3 &= 0, \\ y_4 &= 0, \\ y_5 &= 1. \end{aligned}$$

This results in quadratic equations, which the free parameter has to satisfy. In our case the free parameter is x_5 . Some of the resulting equations are trivial, but others are $x_5^2 = 0$. From this we conclude that $x_5 = 0$ and find the remaining plaintext from (2.16) to (2.19).

Linearization Equations for Multiple-Branch MI

Using the notation of the multiple-branch case discussed above, it is evident we have the following theorem.

Theorem 2.3.4. *Let \mathcal{L} be the space of linearization equations for a given implementation of MI and fix a ciphertext $(y'_1, \dots, y'_n) \in k^n$. Let*

$Y' = \phi^{-1}(y'_1, \dots, y'_n)$ and define $\mathcal{L}_{Y'}$ to be the space of linear equations (in the plaintext variables x_1, \dots, x_n) obtained by substituting in y'_j in for y_j (for $j = 1, \dots, n$) in every element of \mathcal{L} . Then with probability

$$\frac{(q^{n_1} - 1)(q^{n_2} - 1) \cdots (q^{n_b} - 1)}{q^n}$$

$\dim_k \mathcal{L}_{Y'}$ is at least

$$n - \sum_{i=1}^b \gcd(n_i, \theta_i) \geq \frac{2n}{3}$$

Therefore the linearization attack for the single-branch case can also be applied to the multiple-branch case. Additionally, there are refined methods suggested by Patarin [Patarin, 2000] that improve the efficiency of the algorithm where one separates the branches before attacking the system.

From a mathematical point view one can see that it is possible to separate the different branches using the idea of finding a common invariant subspace. This idea was pursued in [Felke, 2005] for the more general case of multi-branch HFE.

Remark 2.3.2. *It is not difficult to see that the attack of Kipnis-Shamir [Kipnis and Shamir, 1999] on the HFE cryptosystem can also be used to attack the Matsumoto-Imai cryptosystem. In this case one can actually recover the private key, and it applies to both single- and multiple-branch cases. One can also see that the linearization attack can be viewed as the prototype and the origin of the XL algorithm for solving polynomial equations.*

2.4 Another Attack on Matsumoto-Imai

In this section, we will present an attack that is an extension of the Kipnis-Shamir attack on HFE for use against the Matsumoto-Imai cryptosystem. Unlike the linearization attack, this attack will allow us to recover the private key. This attack has not been published before, though it is probably known to the experts in this area. The importance of this new approach is that it may lead to a new attack on MI-Minus, which then can be used to attack Sflash^{v2}.

The key idea of the Kipnis-Shamir attack on HFE is to attack the problem from its origin. The constructions of MI and HFE are based on the idea that we can construct a map on a k -vector space from a map on an extension field. Their idea was to use the structure of the map on the extension field to design the attack on the k -vector space mapping.

With this point of view, if $\bar{F} : k^n \longrightarrow k^n$ is a given Matsumoto-Imai public key mapping, then the first step of the attack is to lift \bar{F} back to a map over K ; i.e., we must study $\phi^{-1} \circ \bar{F} \circ \phi$, in order to use the underlying algebraic structures from the extension field, not the vector space over the small field.

To simplify the exposition we assume that $q > 2$ and that L_1, L_2 are linear instead of affine, in effect ignoring the constant terms. In other words, we assume that the $\bar{f}_1, \dots, \bar{f}_n$ are degree two homogeneous polynomials in $k[x_1, \dots, x_n]$. Also, we assume that we know the field K and hence the map $\phi : K \longrightarrow k^n$. If we do not have this information, then we will produce L'_1, L'_2 and F' such that $\bar{F} = L'_1 \circ F' \circ L'_2$. We now justify this claim.

As before, the legitimate user picks an degree n irreducible polynomial $g(x) \in k[x]$ in order to construct $K = k[x]/g(x)$ and $\phi : K \longrightarrow k^n$. Suppose the attacker has chosen another degree n irreducible polynomial $h(y) \in k[y]$ and constructs $K' = k[y]/h(y)$ and $\psi : K' \longrightarrow k^n$. Of course, K and K' are isomorphic, and in fact, k -linear field isomorphisms exist between K and K' . Let $\alpha(y) \in K'$ be such that

$$g(\alpha(y)) = 0 \bmod h(y),$$

and let $\iota : K \longrightarrow K'$ be defined by

$$\iota(p(x)) = p(\alpha(y)) \bmod h(y),$$

for $p(x) \in K$. It is easy to check that ι is a k -linear field isomorphism between K and K' .

Observe that

$$\begin{aligned} \bar{F} &= L_1 \circ F \circ L_2 \\ &= L_1 \circ (\phi \circ \bar{F} \circ \phi^{-1}) \circ L_2 \\ &= L_1 \circ \phi \circ (\iota^{-1} \circ \iota) \circ \bar{F} \circ (\iota^{-1} \circ \iota) \circ \phi^{-1} \circ L_2 \\ &= (L_1 \circ \phi \circ \iota^{-1}) \circ (\iota \circ \bar{F} \circ \iota^{-1}) \circ (\iota \circ \phi^{-1} \circ L_2). \end{aligned}$$

Define $M_1 : K' \longrightarrow k^n$, $\tilde{F}' : K' \longrightarrow K'$, and $M_2 : k^n \longrightarrow K'$ by

$$\begin{aligned} M_1 &= L_1 \circ \phi \circ \iota^{-1} \\ \tilde{F}' &= \iota \circ \bar{F} \circ \iota^{-1} \\ M_2 &= \iota \circ \phi^{-1} \circ L_2. \end{aligned}$$

Observe that M_1 and M_2 are k -linear vector space isomorphisms, and that

$$\begin{aligned}\tilde{F}'(X) &= \iota \left((\iota^{-1}(X))^{q^\theta+1} \right) \\ &= \iota \left(\iota^{-1} \left(X^{q^\theta+1} \right) \right) \\ &= X^{q^\theta+1} \\ &= \tilde{F}(X).\end{aligned}$$

We consider whether or not there exists L'_1 and L'_2 such that

$$\bar{F} = L'_1 \circ \psi \circ \tilde{F}' \circ \psi^{-1} \circ L'_2,$$

or equivalently, for a given M_1 and M_2 , whether or not there exists L'_1 and L'_2 such that

$$\begin{aligned}L'_1 \circ \psi &= M_1 \\ \psi^{-1} \circ L'_2 &= M_2.\end{aligned}$$

Solving for L'_1 and L'_2 we see that

$$\begin{aligned}L'_1 &= M_1 \circ \psi^{-1} \\ L'_2 &= \psi \circ M_2.\end{aligned}$$

The attacker cannot know ι , and thus cannot know M_1 and M_2 . Therefore the attacker cannot know which L'_1 and L'_2 will be obtained. However, if some L'_1 and L'_2 can be found, then these are just as useful as the original L_1 and L_2 for the attack since we can then easily invert the map F anyway. Therefore it does not really matter if the attacker knows the extension field K or not, and thus, there is no advantage in hiding the field structure of K . From now on, we assume that we know the field K .

Now, from Lemma A.0.2 in Appendix A, we know that

$$\phi^{-1} \circ \bar{F} \circ \phi(X) = \sum_{i=0}^{n-1} \sum_{j=0}^{i-1} A_{ij} X^{q^i+q^j}, \quad (2.20)$$

for some $A_{ij} \in K$. We also know that

$$\begin{aligned}\phi^{-1} \circ \bar{F} \circ \phi &= \phi^{-1} \circ (L_1 \circ F \circ L_2) \circ \phi \\ &= \phi^{-1} \circ (L_1 \circ (\phi \circ \tilde{F} \circ \phi^{-1}) \circ L_2) \circ \phi \\ &= (\phi^{-1} \circ L_1 \circ \phi) \circ \tilde{F} \circ (\phi^{-1} \circ L_2 \circ \phi),\end{aligned} \quad (2.21)$$

where the parentheses are added to show the composition is of three maps defined on K . In particular, we know \tilde{F} and we can construct $\phi^{-1} \circ \tilde{F} \circ \phi$ from the known \tilde{F} . Our attack will focus on using the properties of these known maps to find both of the unknown linear maps $\phi^{-1} \circ L_1 \circ \phi$ and $\phi^{-1} \circ L_2 \circ \phi$. In particular, we will study

$$(\phi^{-1} \circ L_1^{-1} \circ \phi) \circ (\phi^{-1} \circ \tilde{F} \circ \phi) = \tilde{F} \circ (\phi^{-1} \circ L_2 \circ \phi),$$

and the properties of the functions in this formula.

From Lemma A.0.1 in Appendix A we have the following equations:

$$\phi^{-1} \circ L_1 \circ \phi(X) = \sum_{j=0}^{n-1} L_{1j} X^{q^j} \quad (2.22)$$

$$\phi^{-1} \circ L_1^{-1} \circ \phi(X) = \sum_{j=0}^{n-1} L_{1j}^{-1} X^{q^j} \quad (2.23)$$

$$\phi^{-1} \circ L_2 \circ \phi(X) = \sum_{j=0}^{n-1} L_{2j} X^{q^j}. \quad (2.24)$$

where $L_{ij} \in K$. Our attack comes down to finding the L_{ij} , from which we can then construct L_1 and L_2 .

Remark 2.4.1. *We make special note that the notation L_{1j}^{-1} represents the coefficient of X^{q^j} in the polynomial representation of L_1^{-1} . This is to be distinguished from $(L_{1j})^{-1}$, the multiplicative inverse of the coefficient of X^{q^j} in the polynomial representation of L_1 . In general these two notations will not refer to the same value in K . All other exponent notations will be written as usual without parentheses.*

Now for any polynomial $G(X) \in K[X]$ of the form

$$G(X) = \sum_{i=0}^{n-1} \sum_{j=0}^i G_{ij} X^{q^i + q^j},$$

we can associate a unique $n \times n$ symmetric matrix G defined by

$$[G]_{ij} = \begin{cases} 2G_{ii} & \text{if } i = j; \\ G_{ij} & \text{if } i > j; \\ G_{ji} & \text{if } i < j. \end{cases}$$

Note that this matrix is such that

$$G(X + Y) - G(X) - G(Y) = \mathbf{x} G \mathbf{y}^T,$$

where $\mathbf{x} = (X, X^q, \dots, X^{q^{n-1}})$ and $\mathbf{y} = (Y, Y^q, \dots, Y^{q^{n-1}})$. We make a special note here that the index of the rows and column range in $0, \dots, n-1$, and not $1, \dots, n$. We also note that because the characteristic of K is two, the entries on the diagonal of G are all zero.

Remark 2.4.2. *The trouble with the case $q = 2$ is that in \bar{F} the square terms and the linear terms are now the same and therefore mixed. But because of the symmetrization process, we realize that these linear terms are only related to the diagonal elements in the matrix, which are annihilated here anyway. Therefore there is no problem with this attack for the case $q = 2$.*

With this correspondence between homogeneous quadratic functions on K and $n \times n$ matrices with entries in K , we will shift from the function point of view to that of matrices. In particular, let \tilde{F} be the matrix associated with \tilde{F} . Then clearly \tilde{F} has only two nonzero entries: $[\tilde{F}]_{00} = 1$ and $[\tilde{F}]_{00} = 1$. To see the basic idea of the attack, we must first understand how the bilinear form behaves if we compose the function by a k -linear function from the left or right. The results are presented in the following two lemmas that deal with how these matrices behave under function composition.

Lemma 2.4.1. *Let $G(X)$ be as defined above, let $S(X) = \sum_{i=0}^{n-1} S_i X^{q^i}$ and let G' be the symmetric matrix associated with $G(S(X))$. Then*

$$G' = W^T G W,$$

where W is an $n \times n$ matrix defined by

$$[W]_{ij} = S_{j-i}^{q^i},$$

and $j - i$ is calculated modulo n .

Proof. We begin by expanding $G(S(X))$:

$$\begin{aligned} G(S(X)) &= \sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} \left(\sum_{l=0}^{n-1} S_l X^{q^l} \right)^{q^u + q^v} \\ &= \sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} \left(\sum_{l=0}^{n-1} S_l X^{q^l} \right)^{q^u} \left(\sum_{l=0}^{n-1} S_l X^{q^l} \right)^{q^v} \\ &= \sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} \left(\sum_{l=0}^{n-1} S_l^{q^u} X^{q^{l+u}} \right) \left(\sum_{l=0}^{n-1} S_l^{q^v} X^{q^{l+v}} \right) \end{aligned}$$

$$\begin{aligned}
&= \sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} \left(\sum_{i=0}^{n-1} S_{i-u}^{q^u} X^{q^i} \right) \left(\sum_{j=0}^{n-1} S_{j-v}^{q^v} X^{q^j} \right) \\
&= \sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} \left(\sum_{i=0}^{n-1} \sum_{j=0}^{n-1} S_{i-u}^{q^u} S_{j-v}^{q^v} X^{q^i+q^j} \right) \\
&= \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \left(\sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} S_{i-u}^{q^u} S_{j-v}^{q^v} \right) X^{q^i+q^j} \\
&= \sum_{i=0}^{n-1} \sum_{j=0}^i \left(\sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} \left(S_{i-u}^{q^u} S_{j-v}^{q^v} + S_{j-u}^{q^u} S_{i-v}^{q^v} \right) \right) X^{q^i+q^j} \\
&\quad - \sum_{w=0}^{n-1} \left(\sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} S_{w-u}^{q^u} S_{w-v}^{q^v} \right) X^{2q^w}.
\end{aligned}$$

Thus the coefficient of $X^{q^i+q^j}$ for $i > j$ is

$$\sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} \left(S_{i-u}^{q^u} S_{j-v}^{q^v} + S_{j-u}^{q^u} S_{i-v}^{q^v} \right).$$

This is the same as $[G']_{ij}$ for $i > j$, since:

$$\begin{aligned}
[G']_{ij} &= [W^T G W]_{ij} = \sum_{u=0}^{n-1} [W^T]_{iu} [G W]_{uj} \\
&= \sum_{u=0}^{n-1} [W]_{ui} \left(\sum_{v=0}^{n-1} [G]_{uv} [W]_{vj} \right) \\
&= \sum_{u=0}^{n-1} S_{i-u}^{q^u} \left(\sum_{v=0}^{n-1} [G]_{uv} S_{j-v}^{q^v} \right) \\
&= \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} [G]_{uv} S_{i-u}^{q^u} S_{j-v}^{q^v} \\
&= \sum_{u=0}^{n-1} \sum_{v=0}^u \left([G]_{uv} S_{i-u}^{q^u} S_{j-v}^{q^v} + [G]_{vu} S_{i-v}^{q^v} S_{j-u}^{q^u} \right) - \sum_{l=0}^{n-1} [G]_{ll} S_{i-l}^{q^l} S_{j-l}^{q^l} \\
&= \sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} \left(S_{i-u}^{q^u} S_{j-v}^{q^v} + S_{i-v}^{q^v} S_{j-u}^{q^u} \right).
\end{aligned}$$

□

Lemma 2.4.2. *Let $G(X)$ and $S(X)$ be defined as in Lemma 2.4.1. Define G'' to be the symmetric matrix associated with $S(G(X))$. Then*

$$G'' = \sum_{l=0}^{n-1} S_l G_l,$$

where G_l is the $n \times n$ matrix defined by

$$[G_l]_{ij} = G_{i-l, j-l}^{q^l},$$

with both $i-l$ and $j-l$ calculated modulo n .

Proof. As with Lemma 2.4.1, we expand $G(S(X))$:

$$\begin{aligned} G(S(X)) &= \sum_{l=0}^{n-1} S_l \left(\sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv} X^{q^u+q^v} \right)^{q^l} \\ &= \sum_{l=0}^{n-1} S_l \left(\sum_{u=0}^{n-1} \sum_{v=0}^u G_{uv}^{q^l} X^{q^{u+l}+q^{v+l}} \right) \\ &= \sum_{l=0}^{n-1} S_l \left(\sum_{i=0}^{n-1} \sum_{j=0}^i G_{i-l, j-l}^{q^l} X^{q^i+q^j} \right) \\ &= \sum_{i=0}^{n-1} \sum_{j=0}^i \left(\sum_{l=0}^{n-1} S_l G_{i-l, j-l}^{q^l} \right) X^{q^i+q^j}. \end{aligned}$$

Thus the coefficient of $X^{q^i+q^j}$ for $i > j$ is

$$\sum_{l=0}^{n-1} S_l G_{i-l, j-l}^{q^l}.$$

This is the same as $[G'']_{ij}$ for $i > j$, since:

$$[G'']_{ij} = \sum_{l=0}^{n-1} S_l [G_l]_{ij} = \sum_{l=0}^{n-1} S_l G_{i-l, j-l}^{q^l}.$$

□

Suppose that \tilde{F}' is the matrix associated with $\tilde{F} \circ (\phi^{-1} \circ L_2 \circ \phi)$. Then from Lemma 2.4.1 we see that

$$\tilde{F}' = L_2^T \tilde{F} L_2, \tag{2.25}$$

where the $n \times n$ matrix L_2 is defined by

$$[L_2]_{ij} = L_{2j-i}^{q^i}. \quad (2.26)$$

Now suppose that \bar{F} is the matrix associated with $\phi^{-1} \circ \bar{F} \circ \phi$, and that \bar{F}'' is the matrix associated with $(\phi^{-1} \circ L_1^{-1} \circ \phi) \circ (\phi^{-1} \circ \bar{F} \circ \phi)$. Then from Lemma 2.4.2 we see that

$$\bar{F}'' = \sum_{l=0}^{n-1} L_{1l}^{-1} \bar{F}_l, \quad (2.27)$$

where

$$[\bar{F}_l]_{ij} = [\bar{F}]_{i-l, j-l}^{q^l}. \quad (2.28)$$

However, we have seen that

$$(\phi^{-1} \circ L_1^{-1} \circ \phi) \circ (\phi^{-1} \circ \bar{F} \circ \phi) = \tilde{F} \circ (\phi^{-1} \circ L_2 \circ \phi), \quad (2.29)$$

and hence

$$\tilde{F}' = M = \bar{F}'', \quad (2.30)$$

where M denotes the common value of \tilde{F}' and \bar{F}'' .

Clearly the matrix \tilde{F} has rank equal to two. Since L_2 is invertible, we see that $M = L_2^T \tilde{F} L_2$ has rank equal to two as well. But this means that the K -linear combination

$$M = \sum_{l=0}^{n-1} L_{1l}^{-1} \bar{F}_l$$

of the n known matrices $\bar{F}_0, \dots, \bar{F}_{n-1}$ has rank two, a condition we can use to find the values of L_{1l}^{-1} . In fact, this is a so-called “MinRank” problem.

Definition 2.4.1. (*MinRank Problem*) Given $n \times n$ matrices A_1, \dots, A_m over a finite field K and $r < n$, find a non-trivial linear combination of

$$A = \alpha_1 A_1 + \dots + \alpha_m A_m$$

such that the rank of A is less than or equal to r .

The general MinRank problem has been studied by Shallit, Frandsen and Buss [Shallit et al., 1996], among others. It generalizes the so-called “Rank Distance Coding” problem posed by Gabidulin [Gabidulin, 1985], which has been studied in [Stern and Chabaud, 1996; Chen, 1996]. This problem is a generalization of the “Minimal Weight” problem of error correcting codes [Berlekamp et al., 1978]. The general MinRank problem

was proven to be NP-complete in [Shallit et al., 1996] for the case where $r = n - 1$, which in this case corresponds to the problem of finding a linear combination of A_1, \dots, A_m which is singular.

Their proof uses the technique of writing a given set of multivariate equations as an instance of MinRank. This result can be extended to other cases like $r = n - 2, n - 3, \dots$, however MinRank is not too hard when r is very small, as is our case.

The approach of Kipnis and Shamir is to use a new relinearization method to solve this problem. Later, Courtois [Courtois, 2001] proposed a more standard and straightforward method to solve this problem that originated from an idea of Coppersmith, Stern and Vaudenay [Coppersmith et al., 1997].

In the most general case, we treat the A_1, \dots, A_m as known, and the $\alpha_1, \dots, \alpha_m$ as variables. If $A = \alpha_1 A_1 + \dots + \alpha_m A_m$ is to have rank r , then each $(r + 1) \times (r + 1)$ submatrix minor must be equal to zero. This means that each $(r + 1) \times (r + 1)$ submatrix yields a total degree $r + 1$ polynomial equation in the m variables $\alpha_1, \dots, \alpha_m$.

In the case under consideration we have $r = 2$. We also know that the $A_l = \bar{F}_l$ are symmetric with diagonal entries equal to zero. This means that the number of nonzero degree three polynomials in the variables $L_{10}^{-1}, \dots, L_{1n-1}^{-1}$ is $\binom{n}{3}(\binom{n}{3} - 1)/2$, where the equation obtained by choosing indices i_1, i_2, i_3 for the rows and j_1, j_2, j_3 for the columns is the same as the equation gotten by choosing indices j_1, j_2, j_3 for the rows and i_1, i_2, i_3 for the columns, and we discard the trivial equations gotten by taking $i_1 = j_1, i_2 = j_2$, and $i_3 = j_3$.

Since the equations are homogeneous, solutions should be thought of in the projective space of K^n . This means that if we find a solution vector $(L_{10}^{-1}, \dots, L_{1n-1}^{-1})$, then $(\alpha L_{10}^{-1}, \dots, \alpha L_{1n-1}^{-1})$ will also be a solution vector for any nonzero $\alpha \in K$. We may as well then take $L_{10}^{-1} = 1$ and substitute this into all the equations to arrive at a system of $\binom{n}{3}(\binom{n}{3} - 1)/2$ degree three equations in the $n - 1$ variables $L_{11}^{-1}, \dots, L_{1n-1}^{-1}$, which we expect will be easy to solve [Courtois, 2001].

At this point we have $\phi^{-1} \circ L_1^{-1} \circ \phi$, and thus L_1 , so we still need to find L_2 . Along the way we have found $M = \tilde{F}' = L_2^T \tilde{F} L_2$, which we will now use to find L_2 . We have two ways to proceed. First, if \tilde{F} is easily inverted (i.e., if the q -Hamming weight degree of $\tilde{F}(X) = X^t$ is relatively small), then we can directly compute $\phi^{-1} \circ L_2 \circ \phi$, and hence, L_2 , from (2.29). Otherwise, we proceed as did Kipnis and Shamir.

Let u_1, \dots, u_{n-D} be a basis of the left kernel of M , where D is the rank of M which we expect to be two. This means that for $i = 1, \dots, n - D$

we have

$$0 = \mathbf{u}_i \mathbf{M} = \mathbf{u}_i \mathbf{L}_2^T \tilde{\mathbf{F}} \mathbf{L}_2.$$

The invertibility of \mathbf{L}_2 implies that

$$0 = \mathbf{u}_i \mathbf{L}_2^T \tilde{\mathbf{F}},$$

and so, because of the special form of $\tilde{\mathbf{F}}$, we know that

$$(0, a_1, a_2, \dots, a_{\theta-1}, 0, a_{\theta+1}, \dots, a_{n-1}) = \mathbf{u}_i \mathbf{L}_2^T,$$

for some $a_1, \dots, a_{\theta-1}, a_{\theta+1}, \dots, a_{n-1} \in K$. Since the \mathbf{u}_i are known, we evidently have $2(n - D)$ linear equations in the n^2 entries of \mathbf{L}_2^T (or equivalently \mathbf{L}_2) by taking the dot product of \mathbf{u}_i with the 1st and θ th columns of \mathbf{L}_2^T , for $i = 1, \dots, n - D$. In fact, the equations are of the form

$$\begin{aligned} \sum_{j=0}^{n-1} u_{ij} L_{2j} &= 0 \\ \sum_{j=0}^{n-1} u_{ij} L_{2j-\theta}^{q^\theta} &= 0. \end{aligned}$$

The first equation is linear in the variables L_{2j} . The second equation can be transformed into a linear equation by raising both sides to the $q^{n-\theta}$ power, yielding

$$\begin{aligned} 0 &= \left(\sum_{j=0}^{n-1} u_{ij} L_{2j-\theta}^{q^\theta} \right)^{q^{n-\theta}} \\ &= \sum_{j=0}^{n-1} u_{ij}^{q^{n-\theta}} \left(L_{2j-\theta}^{q^\theta} \right)^{q^{n-\theta}} \\ &= \sum_{j=0}^{n-1} u_{ij}^{q^{n-\theta}} L_{2j-\theta}^{q^n} \\ &= \sum_{j=0}^{n-1} u_{ij}^{q^{n-\theta}} L_{2j-\theta}. \end{aligned}$$

Thus we have $2(n - D)$ equations

$$\sum_{j=0}^{n-1} u_{ij} L_{2j} = 0$$

$$\sum_{j=0}^{n-1} u_{i+j\theta}^{q^{n-\theta}} L_{2j} = 0,$$

in the n unknowns $L_{20}, L_{21}, \dots, L_{2n-1}$. Assuming these equations are linearly independent, and that $2(n - D) \geq n$, or equivalently $D \leq n/2$, we will be able to solve this system and finally obtain $\phi^{-1} \circ L_2 \circ \phi$, and thus L_2 .

For more details of this attack, including time and memory complexities, the interested reader should check the related HFE case in [Courtois, 2001].

2.5 Matsumoto-Imai Variants

Two methods have been proposed to improve the security of the Matsumoto-Imai cryptosystem. One is called the “Minus” method, and is designed to resist the linearization attacks proposed by Patarin. The other is called the “Plus” method, and is used to make a cipher injective, thus enabling us to decrypt the ciphertext. Among all the Matsumoto-Imai variants proposed for practical use, the most successful is the Minus variant Sflash^{v2}.

The Minus Method

The Minus method was first suggested in [Shamir, 1993] and discovered independently by Patarin and Matsumoto. This method was utilized by Patarin and his collaborators in [Patarin et al., 1998] and elsewhere. As we will see in the case of Matsumoto-Imai, the application of this method clearly eliminates the possibility of the linearization equation attack, if the Minus number r is not too small.

The Minus method consists of deleting a few, say r , polynomial components from a given multivariate public key. For example, suppose $\bar{F} : k^n \rightarrow k^l$ is a public key cryptosystem with polynomial components $\bar{f}_1, \dots, \bar{f}_l \in k[x_1, \dots, x_n]$. In most cases we have $l = n$, but the Minus method can also be used in other cases. Once we apply the Minus method to \bar{F} , for example by deleting the last r components, we will have a new map $\bar{F}^- : k^n \rightarrow k^{l-r}$ defined by

$$\bar{F}^-(x_1, \dots, x_n) = (\bar{f}_1, \dots, \bar{f}_{l-r}). \quad (2.31)$$

The cryptosystem for signatures is, in general, set as follows.

The Public Key

The public key includes:

- 1.) The field structure of k ;
- 2.) The set of polynomials: $(\bar{f}_1, \dots, \bar{f}_{l-r}) \in k[x_1, \dots, x_n]$.

The Private Key

The private key is the same as in the original cryptosystem.

The Signing Process

The document (or its hash value) is $Y'^{-} = (y'_1, \dots, y'_{n-r})$, a vector in k^{n-r} . A legitimate user first chooses (or produces in some way) $n - r$ random elements y'_{n-r+1}, \dots, y'_n in k , which are appended to Y'^{-} to produce $Y' = (y'_1, \dots, y'_n)$ in k^n . Then

$$X' = (x'_1, \dots, x'_n) = \bar{F}^{-1}(Y'),$$

is calculated using the same decryption process as in the original cryptosystem. Finally, X' is the signature of the document Y'^{-} .

The Verifying Process

Anyone who receives the document Y'^{-} and its signature X' first obtains the public key and checks if indeed

$$(\bar{f}_1(X'), \dots, \bar{f}_{l-r}(X')) = Y'^{-}.$$

If equality holds, then the signature is accepted as legitimate, otherwise it is rejected.

In the signing process it is very important that the appended values y'_{n-r+1}, \dots, y'_n are kept secret, otherwise they could be used to recover the missing polynomials to attack the systems as was shown in [Okeya et al., 2005].

The Minus method is particularly useful for converting an encryption scheme (which must be one-to-one) into a signature scheme since we no longer need injectivity. The security of this family of signature schemes is based on the assumption that to solve such a set of $l - r$ nonlinear equations in n variables is very difficult.

In order to illustrate a signature scheme we continue with the toy example, which we used to show how the linearization equation attack works. This time only the polynomials (2.10) to (2.13) are made public, that is (2.14) is hidden and not part of the public key.

The person signing a document has the secret key and with it the linear transformations or their inverses:

$$L_1^{-1}(y_1, y_2, y_3, y_4, y_5) = \begin{pmatrix} \alpha^2 & \alpha & \alpha^2 & 1 & 1 \\ 0 & 0 & \alpha^2 & \alpha^2 & 1 \\ \alpha & \alpha & 1 & \alpha^2 & 1 \\ \alpha & \alpha^2 & 0 & \alpha & 1 \\ 0 & 0 & \alpha & \alpha & 1 \end{pmatrix} \begin{pmatrix} y_1 - \alpha^2 \\ y_2 - \alpha^2 \\ y_3 - 0 \\ y_4 - 1 \\ y_5 - 0 \end{pmatrix}, \quad (2.32)$$

$$L_2^{-1}(y_1, y_2, y_3, y_4, y_5) = \begin{pmatrix} \alpha^2 & 1 & \alpha^2 & 0 & 1 \\ \alpha & 1 & \alpha & 1 & \alpha \\ 0 & \alpha^2 & \alpha & \alpha & 0 \\ \alpha & 1 & 1 & \alpha & \alpha \\ 0 & 1 & 0 & \alpha & \alpha^2 \end{pmatrix} \begin{pmatrix} y_1 - 1 \\ y_2 - 0 \\ y_3 - \alpha^2 \\ y_4 - \alpha^2 \\ y_5 - \alpha^2 \end{pmatrix}. \quad (2.33)$$

Also available for the signing process is $\theta = 3$ of the Matsumoto-Imai map, which gives $\tilde{F}^{-1}(X) = X^{362}$, and the irreducible polynomial $g(x) = x^5 + x^3 + x + \alpha^2$.

Assume that the document (plaintext) to be signed is

$$(\alpha^2, \alpha, \alpha^2, 0).$$

As mentioned above, the additional value should be chosen at random. In our toy example there are only four possibilities for y_5 , and we will display them all

Y' (Document)	X' (Signature)
$(\alpha^2, \alpha, \alpha^2, 0, 0) \implies$	$(0, \alpha, \alpha, 0, \alpha^2),$
$(\alpha^2, \alpha, \alpha^2, 0, 1) \implies$	$(1, 1, \alpha, \alpha, \alpha),$
$(\alpha^2, \alpha, \alpha^2, 0, \alpha) \implies$	$(1, 0, 1, 1, \alpha)$
$(\alpha^2, \alpha, \alpha^2, 0, \alpha^2) \implies$	$(\alpha^2, 1, 1, 0, \alpha^2)$

Any of these signatures, say the first one with $x_1 = 0$, $x_2 = \alpha$, $x_3 = \alpha$, $x_4 = 0$, and $x_5 = \alpha^2$, together with the public key (2.10) to (2.13) will verify that the signature is valid, since we find

$$(y_1, y_2, y_3, y_4) = (\alpha^2, \alpha, \alpha^2, 0).$$

If the four polynomials of the public key are used for an attack via the linearization equation, the attacker would see that $\dim_k \mathcal{L}_{Y'} = 1$ and would only find the equation

$$x_1 = \alpha^2 x_2 + \alpha x_3 + \alpha^2 x_4 + \alpha x_5 + \alpha^2,$$

a relationship satisfied by any of the four signatures. This is not enough to forge a signature. In general, when r becomes larger the linearization equations for the Minus cryptosystem disappear completely.

Flash and Sflash

The New European Schemes for Signatures, Integrity, and Encryption project (NESSIE) within the Information Society Technologies Programme of the European Commission made its final selections for cryptographic primitives at the beginning of 2004 after an evaluation process of more than two years [NESSIE, 1999]. Sflash^{v2}, a fast multivariate signature scheme, was selected by NESSIE as a security standard for use in low-cost smart cards. Sflash^{v2} is called Flash by NESSIE. The initial submission Sflash^{v1} was flawed, as a way was found to break it [Gilbert and Minier, 2002]. The flaw was due to the choice of $GF(2)$ for the field elements. It had been deliberately chosen to minimize the size of the public key. In any case it was not a fatal flaw and it could be corrected easily by choosing $GF(2^7)$ as the field elements in Sflash^{v2} [Patarin et al., 2001; Akkar et al., 2003]. The new version has a signature length of 259 bits and a public key of 15 KBytes.

The authors of the submission claimed that Sflash^{v2} is the fastest signature scheme in the world, and is the only digital signature scheme that can be used in practice for smart cards. Later, due to additional security concerns, the designers of Sflash recommended a new version called Sflash^{v3} [Courtois et al., 2003b], which is essentially Sflash^{v2} with a longer signature. Sflash^{v3} has a signature length of 469 bits and a public key of 112 KBytes. Later, the designers discovered that their security concerns are unfounded and so Sflash^{v2} is again recommended [Courtois, 2004]. At this point it seems that Sflash^{v2}, and with it Flash, should be considered secure.

For ease of exposition we give the basic implementation of Sflash^{v2}. The reader is referred to [Akkar et al., 2003] for technical details. Sflash is a Matsumoto-Imai Minus variant and it uses the single-branch map \bar{F} as given in (2.1) with $\theta = 11$.

Furthermore, Sflash uses $n = 37$ and $r = 11$ so that $\bar{F}^- : k^{37} \longrightarrow k^{26}$ is defined by

$$\bar{F}^-(x_1, \dots, x_n) = (\bar{f}_1, \dots, \bar{f}_{n-r}),$$

where $\bar{f}_1, \dots, \bar{f}_{26} \in k[x_1, \dots, x_{37}]$. The Sflash scheme has the following structure.

Public Key

The following information can be made public, and is needed in order to verify a given Sflash signature:

- 1.) The field $k = GF(2^7)$, including its additive and multiplicative structure. In particular, $k = GF(2)[x]/(x^7 + x + 1)$.

2.) The 26 quadratic polynomials $\bar{f}_1, \dots, \bar{f}_{26} \in k[x_1, \dots, x_{37}]$.

Private Key

The following information should be kept private, and is needed in order to generate Sflash signatures:

- 1.) Δ , a randomly chosen 80-bit long secret key;
- 2.) The two invertible affine transformations L_1 and L_2 associated with the Matsumoto-Imai map \bar{F} .

Signature Generation

Let $\psi : k \longrightarrow GF(2)^7$ be the usual vector space isomorphism. The subscripts below refer to the position in the bit string, and “||” denotes the concatenation of bit strings. In order to sign a message M , we execute the following steps:

- 1.) Compute $M1 = \text{SHA-1}(M)$ and $M2 = \text{SHA-1}(M1)$, two 160-bit strings, using the SHA-1 hash function.
- 2.) Let

$$\begin{aligned} V &= M1 || (M2_1, \dots, M2_{22}) = (V_1, \dots, V_{182}) \\ W &= \text{SHA-1}(V || \Delta) = (W_1, \dots, W_{77}). \end{aligned}$$

- 3.) Let

$$\begin{aligned} M'_1 &= \psi^{-1}(V_1, \dots, V_7) \\ M'_2 &= \psi^{-1}(V_8, \dots, V_{14}) \\ &\vdots \\ M'_{26} &= \psi^{-1}(V_{176}, \dots, V_{182}) \\ \\ M'_{27} &= \psi^{-1}(W_1, \dots, W_7) \\ M'_{28} &= \psi^{-1}(W_8, \dots, W_{14}) \\ &\vdots \\ M'_{37} &= \psi^{-1}(W_{71}, \dots, W_{77}). \end{aligned}$$

Finally let $M' = (M'_1, \dots, M'_{37})$.

4.) Calculate the signature S of M by:

$$\begin{aligned}
 S &= \bar{F}^{-1}(M') \\
 &= L_2^{-1} \circ F^{-1} \circ L_1^{-1}(M') \\
 &= L_2^{-1} \circ \phi \circ \tilde{F}^{-1} \circ \phi^{-1} \circ L_1^{-1}(M'). \tag{2.34}
 \end{aligned}$$

The pair (M, S) represents the message M with signature S .

Signature Verification

Given the message-signature pair (M, S) , we can verify the signature by executing the following steps:

1.) Signature verification begins in the same way as the generation. Compute

$$\begin{aligned}
 M1 &= \text{SHA-1}(M), \\
 M2 &= \text{SHA-1}(M1) \\
 V &= M1 || (M2_1, \dots, M2_{22}) = (V_1, \dots, V_{182}).
 \end{aligned}$$

2.) Let

$$\begin{aligned}
 N'_1 &= \psi^{-1}(V_1, \dots, V_7) \\
 N'_2 &= \psi^{-1}(V_8, \dots, V_{14}) \\
 &\vdots \\
 N'_{26} &= \psi^{-1}(V_{176}, \dots, V_{182})
 \end{aligned}$$

and $N' = (N'_1, \dots, N'_{26})$.

3.) If $N' = \bar{F}^{-1}(S)$, then accept the signature S as valid; otherwise reject S .

It is clear that in order to forge a signature for the message M , we need to be able to find a single pre-image of N' under \bar{F}^{-1} ; i.e., find one solution (not necessarily all solutions) to a system of 26 equations in 37 variables. Here the secret key Δ is also very important in terms of security [Okeya et al., 2005]. Even if only this secret key Δ is leaked, one can defeat the system easily by using it to find the missing (Minus) polynomials. Finally, it is not hard to see that in the case of Matsumoto-Imai, the Minus method eliminates the possibility of the linearization equations attack.

As was previously mentioned, the Minus method is only suitable for signature schemes, where we need to find only a single element in the

pre-image (as opposed to a unique pre-image required for encryption). The “Plus” method is one way in which we can modify a Minus scheme for use in encryption.

The Plus Method

The Plus method amounts to adding a few, say s , randomly chosen polynomial components to a given multivariate scheme, and then mixing them into the public key through an invertible affine transformation. Clearly the degree of the Plus polynomials should be chosen to be the same as the underlying scheme. For example, let us suppose that $\bar{F} : k^n \longrightarrow k^l$ is a mapping associated with some multivariate scheme. We append the s randomly chosen polynomials $p_1, \dots, p_s \in k[x_1, \dots, x_n]$ to create a new map $\bar{F}^+ : k^n \longrightarrow k^{l+s}$ defined by

$$\bar{F}^+ = L_3 \circ (\bar{f}_1, \dots, \bar{f}_l, p_1, \dots, p_s), \quad (2.35)$$

where $L_3 : k^{l+s} \longrightarrow k^{l+s}$ is an invertible affine transformation that mixes the Plus polynomials into the system.

We would like to point out that originally the main purpose of the Plus method was not to improve the security of the original scheme associated with \bar{F} , but rather to make the map \bar{F} , which is not injective, into an injective map, so that it can be used for encryption. In other words, if $\bar{F}^{-1}(y'_1, \dots, y'_l)$ has multiple elements (q^r , in the case of Matsumoto-Imai-Minus), then the Plus polynomials can be used to reduce the number of pre-images to a single element if s is big enough. Equivalently, the Plus polynomials can help to differentiate which is the real plaintext from a set of possible candidates. From a mathematical point view, the Plus is a simple method to make a map M , which is not injective, into an injective map M^+ by adding more components (an embedding map). Roughly speaking, each additional Plus polynomial will reduce the probability of having multiple pre-images by a factor of q .

The Plus method does not improve the security of the Matsumoto-Imai public key cryptosystems when it is applied directly. It does nothing substantial to help in resisting the linearization equation attacks. The linearization equations are still there unlike in the case of the Minus method when there are not enough of them.

As an example of combining both the Plus and Minus methods, we now present the Matsumoto-Imai-Plus-Minus public key cryptosystem. Let $\bar{F} : k^n \longrightarrow k^n$ be a polynomial mapping whose components

$$\bar{f}_1, \dots, \bar{f}_n \in k[x_1, \dots, x_n]$$

form the public key of a Matsumoto-Imai public key cryptosystem. Delete the last r polynomials, add s randomly chosen degree two polynomials $p_1, \dots, p_s \in k[x_1, \dots, x_n]$, and define the map $\bar{F}^\pm : k^n \longrightarrow k^m$ by

$$\bar{F}^\pm = L_3 \circ (\bar{f}_1, \dots, \bar{f}_{n-r}, p_1, \dots, p_s) = (\bar{f}_1^\pm, \dots, \bar{f}_m^\pm), \quad (2.36)$$

where $r \leq s$, $m = n - r + s$ and $L_3 : k^m \longrightarrow k^m$ is an invertible affine transformation. The Matsumoto-Imai-Plus-Minus scheme has the following structure.

Public Key

- 1.) The field k including its additive and multiplicative structure;
- 2.) The $m = n - r + s$ degree two polynomials $\bar{f}_1^\pm, \dots, \bar{f}_m^\pm \in k[x_1, \dots, x_n]$.

Private Key

- 1.) The degree two polynomials $p_1, \dots, p_s \in k[x_1, \dots, x_n]$;
- 2.) The three invertible affine transformations L_1 , L_2 , and L_3 .

Encryption

Given a plaintext $(x'_1, \dots, x'_n) \in k^n$, calculate $(y'_1, \dots, y'_m) \in k^m$ with the public polynomials:

$$(y'_1, \dots, y'_m) = \bar{F}^\pm(x'_1, \dots, x'_n).$$

Decryption

To decrypt a message we execute the following steps:

- 1.) Calculate $(z_1, \dots, z_{n-r+s}) = L_3^{-1}(y'_1, \dots, y'_{n-r+s})$.
- 2.) For each $w = (w_1, \dots, w_r) \in k^r$, compute

$$t_w = (t_1, \dots, t_n) = \bar{F}^{-1}(z_1, \dots, z_{n-r}, w_1, \dots, w_r),$$

and define $T = \{(w, t_w) \mid w \in k^r\}$.

- 3.) For each $(w, t_w) \in T$, check if

$$p_i(t_w) = z_{n-r+i}$$

holds for all $i = 1, \dots, s$. Keep each t_w that satisfy this criteria and discard the rest. If s is large enough, we should have only one element left, the plaintext (x'_1, \dots, x'_n) .

Here the Plus method also serves the purpose of improving the security once the map L_3 is applied, since after the random polynomials are

mixed into the system we cannot tell which are the original polynomials from the Matsumoto-Imai cryptosystem. This at least will make it too difficult to use any method that can be applied to the Matsumoto-Imai-Minus cryptosystems directly.

2.6 The Security of the Matsumoto-Imai Variants

Before using either the Plus or Minus method, we must decide how large (or small) the Plus and Minus should be. For security reasons we should not delete too few polynomials (r should not be too small), and for efficiency reasons we should not add too many polynomials (s should not be too big). The resulting problem of how to choose r and s optimally is not completely settled, though there are some results [Patarin et al., 1998], etc. In this section we will concentrate on the security analysis of the Minus variant of Matsumoto-Imai.

Cryptanalysis of Sflash^{v1}

Recall that for Sflash^{v1} the field k is chosen to be $GF(2)$, and in particular $k = GF(2)[x]/(x^7 + x + 1)$. The extension field K is chosen to be $k[x]/r(x)$, where $r(x) = x^{37} + x^{12} + x^{10} + x^2 + 1$ is irreducible in $k[x]$, and we know that $n = 37$, $\theta = 11$ and $r = 11$. The two secret maps $L_1, L_2 : k^n \rightarrow k^n$ are specially chosen in that they are taken from a small subset of invertible affine transformations on k^n whose matrix representations have entries only from the subfield $GF(2)$.

Although we can use Sflash to sign documents from k^{26} , it is not hard to see that due to the special choice of $r(x)$, L_1 and L_2 , the public signature verification polynomials all lie in the polynomial ring $GF(2)[x_1, \dots, x_{37}]$. This reduces the required memory by a factor of seven from what it otherwise would be. On the other hand, it is straightforward to check that the public polynomial components obtained by taking $q' = 2$, $n' = n = 37$ and $\theta' = 3$ (so that the fields are $k' = GF(2)$ and $K' = GF(2^{37})$) will yield exactly those of \bar{F} . This is because

$$3 \equiv 7 \times 11 \pmod{37}.$$

Furthermore, if we delete $r' = r = 11$ polynomials, we have a version of Sflash that is much easier to attack. The strategy of Gilbert and Minier [Gilbert and Minier, 2002] is to find the $GF(2)$ -linear span of the deleted polynomials of this “smaller” version of Sflash. Any subset of eleven linearly independent polynomials from this span can be used with the original public polynomials to calculate signatures in the original Sflash signature scheme.

We may now think of \bar{F}^- as a Matsumoto-Imai map from $GF(2^{37})$ to $GF(2^{26})$. Since $GF(2^{37})$ is a relatively small finite field, we can use

brute force to invert the map \bar{F}^- over $GF(2^{37})$. In other words, for every $Y^- \in GF(2^{26})$ we can efficiently compute the set

$$U_{Y^-} = \{X \in GF(2^{37}) \mid \bar{F}^-(X) = Y^-\},$$

which can be stored for later use during the attack.

The strategy of the attack is to find r additional quadratic polynomials q_1, \dots, q_r of the form

$$q_l(x_1, \dots, x_n) = \sum_{i=1}^n \sum_{j=1}^{i-1} \alpha_{ijl} x_i x_j + \sum_{i=1}^n \beta_{il} x_i, \quad (2.37)$$

where $\alpha_{ijl}, \beta_{il} \in GF(2)$, which together with the $n - r$ public quadratic polynomials from \bar{F}^- will span the same linear space as all of the components of \bar{F} except for some constant shift. This gives us an equivalent Matsumoto-Imai polynomial mapping \bar{F}' that can then be subjected to the linearization attack by Patarin. For a given message we cannot use \bar{F}' to produce the exact same signature as we would obtain by using \bar{F} . However, since the span of the components of \bar{F}' is the same as the span of the components of \bar{F} , we can nevertheless produce valid signatures. In other words, if the legitimate user computes S as the signature of M , then at the end of this attack we will be able to compute S' such that $\bar{F}^-(S) = \bar{F}^-(S')$, and therefore can make a successful forgery of the legitimate signature.

The key step in the attack is the characterization of the coefficients of the $q_l(x_1, \dots, x_n)$ by using the fact that \bar{F} is an invertible map and therefore one-to-one. This allows us to reduce the possible candidates for $q_l(x_1, \dots, x_n)$ from the space of all quadratic functions with coefficients in $GF(2)$ (a space with dimension $n(n-1)/2 + n = 703$) to a much smaller space of dimension $4 \times 37 = 148$. Though this space is still much too large, once we get to this point we will be able to reduce the dimension further to solve our problem.

The First Step of the Attack

We begin by noting that \bar{F} is one-to-one, and therefore for each $Y^- = (y_1, \dots, y_{26}) \in GF(2^{26})$, the set U_{Y^-} will have exactly 2^{11} elements. Moreover, for each q_l of the form in (2.37) we must have

$$\sum_{X \in U_{Y^-}} q_l(X) = 0, \quad (2.38)$$

for $l = 1, \dots, 11$. This also follows from the injectivity of \bar{F} , which implies that exactly half of the elements $X \in U_{Y^-}$ are such that $q_l(X) = 0$,

while the other half are such that $q_l(X) = 1$. Therefore, each U_{Y^-} provides one linear equation in the 703 coefficients of the quadratic function q_l . Generating U_{Y^-} for each Y^- can be done by simply calculating $\bar{F}^-(X)$ for each of the 2^{37} elements $X \in K'$.

According to Gilbert and Minier, it is often only necessary to compute U_{Y^-} for $N = 1000$ (a little more than 703) different Y^- . In any case, the N sets U_{Y^-} can be used to obtain an $N \times 703$ matrix with coefficients in $GF(2)$, whose kernel can be computed. This kernel, which we denote \mathcal{Q} , has dimension $37 \times 4 = 148$, and contains the $GF(2)$ -vector space spanned by the 26 public polynomials and the 11 deleted polynomials (without constant terms). We now explain the appearance of spurious polynomials, polynomials not in the span of the components of \bar{F} . Before we do this, we first need to say a few words about discrete derivatives.

Discrete Derivatives

We consider only the case of a finite field of characteristic two. Let V be a vector space and let g be any function from V to V . The derivative of g with respect to the vector $v \in V$ is then defined to be:

$$\partial_v(g(x)) = g(x) + g(x + v).$$

More generally, if $W = \{v_1, \dots, v_m\}$ is a subset of vectors in V , then the derivative of g with respect to the set of vectors W is defined to be:

$$\begin{aligned} \partial_W(g(x)) &= \partial_{v_1}(\partial_{v_2}(\dots(\partial_{v_m}(g(x)))\dots)) \\ &= \sum_{w \in W'} g(x + w), \end{aligned}$$

where W' is the set of all linear combinations $\alpha_1 v_1 + \dots + \alpha_m v_m$ with $\alpha_1, \dots, \alpha_m \in \{0, 1\}$.

Now suppose W is an m -dimensional subspace of V , and that W has basis $B = \{v_1, \dots, v_m\}$. Then we define the derivative of g with respect to the vector space W as just $\partial_B(g(x))$, though we will abuse notation and write $\partial_W(g(x))$. We note that if V is a $GF(2)$ -vector space, then

$$\partial_W(g(x)) = \sum_{w \in W} g(x + w).$$

Finally, let A be an affine set of dimension m , so that $A = v + W$ for some vector $v \in V$ and m -dimensional subspace W . Then the derivative of g with respect to the affine set A is defined to be $\partial_B(g(x + v))$, where B is any basis of the subspace W . As before, we will abuse notation and write $\partial_A(g(x))$. If V is a $GF(2)$ -vector space, then

$$\partial_A(g(x)) = \sum_{w \in W} g(x + v + w).$$

The following two results about the discrete derivative will be particularly useful when the vector space has an additional ring structure.

Lemma 2.6.1. *Suppose K is a degree n field extension of $GF(2)$, let $g(x)$ be a nonzero polynomial in $K[x]$, and pick any $a \in K$. Then the Hamming weight degree of $\partial_a(g(x))$ is strictly less than the Hamming weight degree of $g(x)$.*

Proof. Since the discrete derivative is clearly additive, it suffices to consider the case of $g(x) = x^l$ for $l \geq 0$. Suppose that there are m nonzero terms in the binary expansion of l :

$$l = 2^{i_1} + 2^{i_2} + \dots + 2^{i_m}.$$

Then

$$\begin{aligned} \partial_a(g(x)) &= g(x) + g(x+a) \\ &= x^l + (x+a)^l \\ &= x^l + (x+a)^{2^{i_1}+2^{i_2}+\dots+2^{i_m}} \\ &= x^l + (x+a)^{2^{i_1}}(x+a)^{2^{i_2}} \dots (x+a)^{2^{i_m}} \\ &= x^l + (x^{2^{i_1}} + a^{2^{i_1}})(x^{2^{i_2}} + a^{2^{i_2}}) \dots (x^{2^{i_m}} + a^{2^{i_m}}) \\ &= x^l + x^{2^{i_1}+2^{i_2}+\dots+2^{i_m}} + \text{lower weight terms} \\ &= 2x^l + \text{lower weight terms} \\ &= 0 + \text{lower weight terms}, \end{aligned}$$

where the last equality holds since the characteristic of K is two. \square

Corollary 2.6.1. *Suppose K is a degree n field extension of $GF(2)$, and let $\phi : K \rightarrow GF(2)^n$ be the usual identification. Pick $g(x) \in K[x]$ of Hamming weight degree d . If A is any m -dimensional affine set in $GF(2)^n$ with $d \leq m$, then*

$$\partial_{\phi^{-1}(A)}(g) = 0.$$

Proof. The proof follows directly from the previous lemma. \square

Spurious Polynomials

Fix $Y^- = (y'_1, \dots, y'_{26})$ and let

$$V_{Y^-} = \{(y_1, \dots, y_{37}) \in GF(2)^{37} \mid (y_1, \dots, y_{26}) = Y^-\},$$

an affine subset of $GF(2)^{37}$. Let Y be any element in V_{Y^-} and suppose $X = (x_1, \dots, x_n)$ satisfies $\bar{F}(X) = Y = (y_1, \dots, y_n)$. If q_l is in the span

of the components of \bar{F} (i.e., $q_l = \sum_{i=1}^n a_i \bar{f}_i$), then we must have that

$$q_l(x_1, \dots, x_n) = q_l(\bar{F}^{-1}(y_1, \dots, y_n)) = \sum_{i=1}^n a_i y_i, \quad (2.39)$$

where the second equality comes from the fact that

$$y_i = \bar{f}_i(\bar{F}^{-1}(y_1, \dots, y_n)),$$

for $i = 1, \dots, n$. In this way we can associate with $q_l(x_1, \dots, x_n)$ a new function

$$\tilde{q}_l(y_1, \dots, y_n) = q_l \circ \bar{F}^{-1}(y_1, \dots, y_n).$$

With this shift in perspective we have

$$\sum_{X \in U_{Y-}} q_l(x_1, \dots, x_n) = \sum_{Y \in V_{Y-}} \tilde{q}_l(y_1, \dots, y_n). \quad (2.40)$$

Since V_{Y-} is an affine subset in $GF(2)^{37}$, the sum $\sum_{X \in U_{Y-}} q_l(x_1, \dots, x_n)$ is now realized as a (discrete) derivative of the function $\tilde{q}_l(y_1, \dots, y_n)$, which is itself a linear function in the y_1, \dots, y_n , provided that $q_l = \sum_{i=1}^n a_i \bar{f}_i$.

Therefore, an equation of the form of (2.38) will be satisfied by any total degree two polynomial $q(x_1, \dots, x_n)$ such that $\tilde{q}(y_1, \dots, y_n) = q \circ \bar{F}^{-1}(y_1, \dots, y_n)$ can be expressed as a polynomial of total degree at most 10 in the y_1, \dots, y_n . Let us now explore how such functions occur.

Let $\tilde{F}_i : K' \rightarrow K'$ be defined by

$$\tilde{F}_i(X) = X^{2^i+1},$$

for $i = 0, \dots, 36$, and let $F_i : k^n \rightarrow k^n$ be defined by

$$F_i = \phi \circ \tilde{F}_i \circ \phi^{-1} \circ L_2 = (f_{i1}, \dots, f_{in}),$$

deviating slightly from the usual notation. Clearly \tilde{F} is \tilde{F}_3 .

Take $Y = \tilde{F}(X) = L_1 \circ F_3(X)$. Then $F_3(X) = L_1^{-1}(Y)$. Also, $\tilde{F}_3^{-1}(X) = X^t$, where $t \equiv (2^3 + 1)^{-1} \pmod{(2^{37} - 1)}$. Therefore, if any quadratic polynomial $q(X)$ (with total degree two in the components x_1, \dots, x_n of X) is equal to a linear combination of the components of some $F_i(X) = (f_{i1}, \dots, f_{in})$, then \tilde{q} can be expressed as a linear combination of the quadratic terms of the 37 $GF(2)$ -components of $F_i \circ F^{-1}$. To see why this is true, consider the following. Assume

$$q(x_1, \dots, x_n) = \sum_{j=1}^n a_j f_{ij},$$

and take

$$L(x_1, \dots, x_n) = \sum_{j=1}^n a_j x_j.$$

We then clearly have

$$q(x_1, \dots, x_n) = L \circ F_i,$$

and thus,

$$\begin{aligned} \sum_{X \in U_{Y-}} q(X) &= \sum_{Y \in V_{Y-}} \tilde{q}(Y) \\ &= \sum_{Y \in V_{Y-}} q \circ \bar{F}^{-1}(Y) \\ &= \sum_{Y \in V_{Y-}} L \circ F_i \circ \bar{F}^{-1}(Y) \\ &= \sum_{Y \in V_{Y-}} L \circ F_i \circ L_2^{-1} \circ \phi \circ \tilde{F}^{-1} \circ \phi^{-1} \circ L_1^{-1}(Y) \\ &= \sum_{Y \in V_{Y-}} L \circ \phi \circ \tilde{F}_i \circ \tilde{F}^{-1} \circ \phi^{-1} \circ L_1^{-1}(Y), \end{aligned}$$

the degree of the last expression in the components of $Y = (y_1, \dots, y_n)$ being bounded above by the Hamming weight of the degree of $\tilde{F}_i \circ \tilde{F}^{-1}$, which is $t(2^i + 1) \bmod (2^{37} - 1)$.

One can easily compute $d_i = t(2^i + 1) \bmod (2^{37} - 1)$ for $i = 0, \dots, 36$ and find that there are exactly four values of i such that the Hamming weight w_i of d_i is at most 10. In particular, we find that:

$$\begin{aligned} d_3 &= 1 = (1)_2 \implies w_3 = 1 \\ d_9 &= 57 = (111001)_2 \implies w_9 = 4 \\ d_{15} &= 3641 = (111000111001)_2 \implies w_{15} = 7 \\ d_{21} &= 233017 = (111000111000111001)_2 \implies w_{21} = 10 \end{aligned}$$

and thus the components of $\tilde{F}_3, \tilde{F}_9, \tilde{F}_{15}$, and \tilde{F}_{21} can all be expressed as functions of degree at most 10 in the components of Y . Therefore any linear combination of these $4 \times 37 = 148$ polynomials will satisfy an equations of the form in (2.38).

The Second Step of the Attack

We must now further characterize the coefficients of the desired $q_i(x)$. We will use the public knowledge we know about \bar{F} to express additional

conditions we can use to determine the $q_i(x)$ completely. Computer experiments confirm that these additional conditions do indeed determine the $q_i(x)$.

Choose a basis for \mathcal{Q} using Gaussian elimination, say $\{q_1, \dots, q_{148}\}$. We need a condition on the γ_i such that $q = \sum \gamma_i q_i(x)$ must belong to the space spanned by $\bar{f}_1, \dots, \bar{f}_n$.

Let $q(x_1, \dots, x_n) \in \mathcal{Q}$. From the condition imposed by (2.38) on q , we see that the total degree of $\tilde{q}(y_1, \dots, y_n)$ cannot be more than 10, and that if $q(x_1, \dots, x_n)$ belongs to the space spanned by $\bar{f}_1, \dots, \bar{f}_n$ then the total degree of $\tilde{q}(y_1, \dots, y_n)$ is 1, as we have seen from (2.39). Thus, if $q(x_1, \dots, x_n)$ is indeed in the space spanned by $\bar{f}_1, \dots, \bar{f}_n$, then for $i = 1, \dots, 148$, the derivative with respect to any 12-dimensional affine set \mathcal{A} of $\tilde{q}_i \tilde{q}$ (whose degree is at most $10 + 1 = 11$) will be zero. On the other hand, if $q(x_1, \dots, x_n)$ does not belong to the space spanned by $\bar{f}_1, \dots, \bar{f}_n$, then the degree of $\tilde{q}_i \tilde{q}$ is expected to be at least $10 + 4 = 14$, due to the fact that the Hamming weight of $t(2^i + 1) \bmod (2^{37} - 1)$ for $i = 9, 15, 21$ are of weight 4, 7, 10, respectively. Therefore we do not expect that the derivative of $\tilde{q}_i \tilde{q}$ will be zero. We are now ready to formulate the desired conditions on the γ_i .

Let $Y^{--} = (y_1, \dots, y_{25}) \in GF(2)^{25}$, and let us denote by $V_{Y^{--}}$ the affine subset of $GF(2)^{37}$

$$V_{Y^{--}} = \{(y_1, \dots, y_{37}) \in GF(2)^{37} \mid (y_1, \dots, y_{25}) = Y^{--}\}.$$

With this notation we have

$$\sum_{Y \in V_{Y^{--}}} \tilde{q}_i(Y) \tilde{q}(Y) = 0.$$

For each $Y^{--} = (y_1, \dots, y_{25})$, define $Y_0^- = (y_1, \dots, y_{25}, 0)$ and $Y_1^- = (y_1, \dots, y_{25}, 1)$, and let $U(Y^{--}) = U_{Y_0^-} \cup U_{Y_1^-}$. The above equation gives rise to a linear equation in the 148 unknown $GF(2)$ -coefficients γ_i of q in the form:

$$\sum_{X \in U(Y^{--})} \sum_{i=1}^{148} \gamma_i q_i(X) q(X) = 0. \quad (2.41)$$

In their computer experiments, Gilbert and Minier actually needed to use only two arbitrary quadratic polynomials, q_1 and q_2 , which allowed them to collect $N' = 200$ (a little more than 148 equations) to obtain a solution space of dimension exactly 37. This completes step two of the attack.

Once this is done we have the space spanned by \bar{f}_i . After picking a basis for this much smaller space, we use the linearization attack to

invert the Sflash public polynomials for any given image. This allows us to forge signatures.

Complexity

The most complex calculation required by the attack above is the exhaustive computation of the 2^{37} values of the public function \bar{F}^- , which is needed to obtain the (at most) $N + 2N'$ sets of 2^{11} pre-images required for the computations of the attack. The computations of Step 1 are the derivation of the $N = 1000$ linear equations in 703 variables and the Gaussian elimination of the resulting $N \times 703$ system, so the complexity of Step 1 is bounded above by $N \times 703 \times 2^{11} + N^3/3 < 2^{32}$. Similarly, the complexity of the derivation of the N' linear equations in 148 variables and the Gaussian elimination of the resulting $N' \times 148$ system in Step 2 is bounded above by 2^{27} . These are far lower than 2^{37} computations of the Sflash^{v1} public functions. We also note that the complexity of the linearization attack is about 2^{27} computations. Therefore the complexity of the entire attack is bounded above by 2^{37} .

The attack presented above is based on the fact that the Sflash^{v1} public function over k^{37} induces a restricted function over the much smaller vector space $GF(2)^{37}$. This attack does not seem to be applicable to more conservative instances of the Matsumoto-Imai-Minus scheme, such as Sflash^{v2}, since a much more efficient method would then have to be found to determine each set of q^r preimages under \bar{F}^- . In this case $q^r = (2^7)^{11} = 2^{77}$, which makes the brute force search for the set of pre-images by Gilbert and Minier above impossible.

Other Attacks on MI-Minus

In [Patarin et al., 1998], a general attack on the Matsumoto-Imai-Minus family was presented. This attack is essentially a differential type of attack where one uses the fact that \bar{F} is an invertible map. The starting point is to use the so-called polar form of \bar{F} given by

$$Q(X, T) = \bar{F}(X + T) - \bar{F}(X) - \bar{F}(T),$$

which in this case is related to bilinear forms of the polynomials components of \bar{F} . If we fix X to be a constant, then the equation above becomes linear in T . This method utilizes the fact that the public key polynomials come from a set of permutation polynomials, which allows us to use the general theory about permutation polynomials and the idea of orthogonal systems of equations [Lidl and Niederreiter, 1997]. Then we may look for a value X such that solution space is of maximum dimension. The basic idea is to use this solution space to find a way to recover the lost (Minus) polynomials and then use again the linearization

equations to break the system. From this we can see that this attack in essence is closely related to the attack by Gilbert and Minier above. We will omit the details of the attack here and refer the readers to the original paper [Patarin et al., 1998].

It is shown that such an attack should have complexity of $O(q^r)$, and therefore it is suggested that q^r should be at least 2^{64} in order to guarantee security against this attack. This attack is also very closely related to the differential attack [Fouque et al., 2005] on PMI [Ding, 2004a], which will be discussed later.

We believe that the new attack on MI in Section 2.4 can also be directly extended to attack the MI-Minus cryptosystem, especially when the Minus number r is small.

Security of MI-Plus-Minus

We believe that the security of MI-Plus-Minus is also still open, since it should be a much harder problem to attack MI-Plus-Minus than MI-Minus in general. Moreover, there is also a problem of how big the Plus can be before additional security concerns arise. In [Patarin et al., 1998], some attacks were suggested for MI-Plus-Minus that are actually prototypes of the XL-family of algorithms. We will leave the details of this discussion for the chapter on general methods for solving systems of polynomial equations.

Related work

First we like to point out that the Matsumoto-Imai cryptosystems we talk about in this chapter should not be confused with some of their other cryptosystems from 1983 [Matsumoto and Imai, 1983]. These were broken in 1984 [Delsarte et al., 1985] and are very different systems from what we study here.

The original ideas of the Matsumoto-Imai cryptosystems were first presented in [Imai and Matsumoto, 1985]. In the 1988 paper, two families of systems are discussed. The other one is the so-called Hidden Matrix (HM) scheme, where the key map uses matrix multiplications, and in particular the square of a matrix. These schemes were defeated by using the same method of linearization equations [Patarin et al., 1998]. In the 1985 paper [Imai and Matsumoto, 1985], there is also another scheme called the “B” scheme, and it was broken in 2001 [Youssef and Gong, 2001] using statistical methods.

In the process of developing a new differential method to attack PMI [Ding, 2004a], Fouque, Granboulan, and Stern also found a new differential attack to break the MI [Fouque et al., 2005].

From [Felke, 2005], we also see that the linearization attack was independently discovered by Dobbertin at the German Information Security Agency in 1993.



<http://www.springer.com/978-0-387-32229-2>

Multivariate Public Key Cryptosystems

Ding, J.; Gower, J.E.; Schmidt, D.S.

2006, XVIII, 260 p., Hardcover

ISBN: 978-0-387-32229-2