On the Security of the KMOV Public Key Cryptosystem

D. Bleichenbacher

Bell Laboratories
700 Mountain Ave.
Murray Hill, NJ 07974
E-mail: bleichen@research.bell-labs.com

Abstract. This paper analyzes the KMOV public key cryptosystem, which is an elliptic curve based analogue to RSA. It was believed that this cryptosystem is more secure against attacks without factoring such as the Håstad-attack in broadcast application. Some new attacks on KMOV are presented in this paper that show the converse. In particular, it is shown that some attacks on RSA which work only when a small public exponent e is used can be extended to KMOV, but with no restriction on e. The implication of these attacks on related cryptosystems are also discussed.

1 Introduction

In 1985, Koblitz and Miller independently proposed new public key cryptosystems based on elliptic curves [9, 16]. These cryptosystems rely on the difficulty to solve the discrete logarithm problem for elliptic curves. Other cryptosystems based on the same problem have been proposed thereafter. We refer to [15] for more information. A more recent overview is [1].

Koyama, Maurer, Okamoto and Vanstone proposed another kind of elliptic curve based cryptosystems [11]. Their schemes are based on the difficulty of factoring large numbers and are similar to RSA and the Rabin scheme. The most practical of these schemes (Type 1) is generally called the KMOV public key cryptosystem, according to the first letters of the author's names. This scheme was the base for a few similar cryptosystems. Demytko proposed a scheme, which uses only one coordinate of a point over an elliptic curve to represent messages and ciphertexts [5]. Koyama proposed a scheme that is based on singular cubic curves [10]. Another closely related cryptosystem proposed by Koyama and Kuwakado in [14].

It is believed that breaking these systems as well as RSA completely is as difficult as factoring. However, there exist a few attacks on RSA which do not require to factor the modulus. Such attacks are sometimes possible when the ciphertexts and some additional information is known, i.e. (i) when some parts of the plaintext is known, (ii) the encryption of the same or related plaintexts is sent to different users (e.g. in a broadcast application) or (iii) when the encryptions of two related plaintexts are sent to the same user.

A few authors have shown that such attacks can be extended to elliptic curve cryptosystems [14, 12, 20, 8]. These attacks are based on division polynomials

whose degree e^2 grows quadraticly with the public parameter e. Because of these results and the more complex structure of KMOV, it is sometimes believed that KMOV is more resistant against this kind of attacks.

In this paper, we present new attacks on KMOV which do not depend on e. In particular, it is shown that the plaintext can be found with high probability (but not always) in each of the following situations:

(i) partially known plaintext: The ciphertext and one half of the plaintext is known.

(ii) broadcast application: 3 encryptions of the same message or 6 encryptions of linearly related messages are known. All messages are encrypted with distinct public keys.

(iii) related messages for the same user: The encryptions of two (linearly) related messages are known. Here, both messages are encrypted with the same public key.

2 Definition of Elliptic curves

This section gives a summary of basic facts about elliptic curves over the field $\mathbb{Z}/(p)$. Let a, b be two integers, such that $4a^3 + 27b^2 \not\equiv 0 \pmod{p}$. By $E_{a,b}(p)$ we denote the group whose elements are given by $\{(x_1, y_1) \in (\mathbb{Z}/(p))^2 : y_1^2 \equiv x_1^3 + ax_1 + b \pmod{p}\} \cup \{\mathcal{O}\}$. By $\mathcal{O}$ we denote the point at infinity, which will also be the neutral element of $E_{a,b}(p)$. The inverse of a point (x_1, y_1) is $(x_1, -y_1)$. The sum $(x_3, y_3) = (x_1, y_1) + (x_2, y_2)$ of two points that are not the inverse of each other can be computed by

$$\lambda \equiv \begin{cases} \dfrac{3x_1^2 + a}{2y_1} \pmod{p} & \text{if } x_1 \equiv x_2 \pmod{p} \\ \dfrac{y_1 - y_2}{x_1 - x_2} \pmod{p} & \text{if } x_1 \not\equiv x_2 \pmod{p} \end{cases}$$

$$x_3 \equiv \lambda^2 - x_1 - x_2 \pmod{p}$$

$$y_3 \equiv \lambda(x_1 - x_3) - y_1 \pmod{p}$$

A multiplication of a point P by an integer t will be denoted by $t \cdot P$

$$t \cdot P = \overbrace{P + \cdots + P}^{t}.$$

Let p, q be two distinct primes and $n = pq$. Then $E_{a,b}(n)$ will be defined by

$$E_{a,b}(n) = E_{a,b}(p) \times E_{a,b}(q)$$

If a point $(x_1, y_1) \in (\mathbb{Z}/(n))^2$ satisfies $y_1^2 \equiv x_1^3 + ax_1 + b \pmod{n}$ then we will associate (x_1, y_1) with the point $((x_1 \bmod p, y_1 \bmod p), (x_1 \bmod q, y_1 \bmod q)) \in E_{a,b}(p) \times E_{a,b}(q)$. Two points represented like this can be added by using the same arithmetic operation as in the definition, however, computed over $\mathbb{Z}/(n)$. The points $(\mathcal{O}, P)$ and $(P, \mathcal{O}) \in E_{a,b}(n)$ can not be represented like this. Finding such a point is, however, very unlikely and would lead to a factorization of n.

3 Description of KMOV

Koyama, Maurer, Okamoto and Vanstone proposed three cryptosystems based on elliptic curves in [11]. We describe their Type 1 scheme here. We will not consider the Type 0 scheme, which seems less practical than the Type 1 scheme, because the order of a general elliptic curve must be computed and the Type 2 scheme, which is a Rabin-type generalization.

The private key of the Type 1 scheme consists of two large primes $p \equiv q \equiv 2 \pmod 3$. The public key consists of the product $n = pq$ and an integer e that is relatively prime to $(p+1)(q+1)$. A message is a pair (m_x, m_y) where $m_x, m_y \in \mathbb{Z}/(n)$. It is encrypted by computing $(c_x, c_y) \equiv e \cdot (m_x, m_y) \pmod n$ over the elliptic curve $E_{0,b}(n)$ where $b \equiv m_y^2 - m_x^3 \pmod n$ is determined by the message. Both values c_x and c_y are sent to the receiver. The receiver can determine the curve over which the message was encrypted (even though this computation is in fact not necessary) from the ciphertext since (c_x, c_y) and (m_x, m_y) are points on the same curve and therefore

$$b \equiv c_y^2 - c_x^3 \equiv m_y^2 - m_x^3 \pmod n. \tag{1}$$

Then, he can decrypt the message by computing $(m_x, m_y) \equiv d \cdot (c_x, c_y) \pmod n$ where $d \equiv e^{-1} \pmod{(p+1)(q+1)}$. This follows from the fact that the order of the curve $E_{0,b}(n)$ divides $(p+1)(q+1)$.

4 The value of a modular polynomial equation of small degree

All RSA-based cryptosystems are based on the difficulty of solving polynomial equations over $\mathbb{Z}/(n)$. No method for solving a univariate equation $f(m) \equiv 0 \pmod n$ for m where $f(x)$ is a polynomial of degree > 1 is known without factoring n. However, if some additional information on a solution m is known then this situation may change. Such situations are described in this section.

Coppersmith describes an algorithm that finds a root of a univariate polynomial if this root is small enough [3]. In particular, he proved the following result.

Theorem 1 (Coppersmith). *Let $f(x)$ be a monic integer polynomial of degree k and N a positive integer of unknown factorization. In time polynomial in $\log N$ and k, we can find all integer solutions m to $f(m) \equiv 0 \pmod N$ with $|m| < N^{1/k}$.*

Coppersmith also discusses the application of his method to general multivariate polynomials [3, Section 3], but we will only describe the implication to polynomials in two variables here. If $f(x, y)$ is a polynomial of total degree k then he showed, that it is often possible to find efficiently a solution m_x, m_y to $f(m_x, m_y) \equiv 0 \pmod N$ if

$$\max(|m_x|, |m_y|) < N^{1/2k-\epsilon}$$

for some $\epsilon > 0$. Even though there is no guarantee that a solution will be found, our experiments have almost always been successful, because the lattice reduction algorithm had found the small lattice vector related to the solution.

Another method that will be used in this paper was discovered by Coppersmith, Franklin, Patarin and Reiter [4]. The authors observed that an unknown value m can often be found when two polynomial equations of small degree $f(m) \equiv g(m) \equiv 0 \pmod{n}$ are known, since $(x - m)$ must divide the gcd of $f(x)$ and $g(x)$ and since $\gcd(f(x), g(x))$ is very likely a linear polynomial. Their attack is practical if the degrees of the polynomials are smaller than about 2^{32}.

Description of an improved algorithm. In this paper, we will study some attacks that are based on the more general problem where only $f(x)$ is a polynomial of small degree and where $g(x)$ is a rational function of large degree that can be computed in a small number of arithmetic steps. For example, if one coordinate is fixed then the encryption function in KMOV defines such a rational function $g(x)$, that can be computed in a small number of operations (i.e. $O(\log(e))$) even though the degree of the function may be large (i.e. e^2).

Even though, the small polynomial equation $f(m) \equiv 0 \pmod{n}$ can generally not be solved, $f(x)$ can be regarded as an implicit representation of m. Using this representation it is possible to perform arithmetic operations on m in almost the same way as arithmetic operations with algebraic numbers are performed. For example, it will be possible to compute an encryption on a message m given implicitly by a small polynomial equation $f(m) \equiv 0 \pmod{n}$.

The first step of our algorithm is a square free factorization on $f(x)$. Thus allows us to assume that $f(x)$ is in fact square free over $\mathbb{Z}[x]/(n)$. In the following, we will perform arithmetic operations in the quotient ring $R = \mathbb{Z}[x]/(n, f(x))$. x will be an implicit representation of m. More generally, any polynomial $h(x) \in R$ will represent $h(m)$ and hence we can define a ring homomorphism $\phi : R \to \mathbb{Z}/(n)$ given by

$$\phi : h(x) \mapsto h(m).$$

Note that ϕ is initially not known explicitly since the solution m is unknown. Note also that ϕ is well defined since $f(m) \equiv 0 \pmod{n}$, i.e. ϕ does not depend on representatives of an equivalence class in R as

$$\phi(h(x) + h'(x)f(x)) \equiv h(m) + h'(m)f(m) \equiv h(m) \equiv \phi(h(x)) \pmod{n}.$$

Let t be the degree of $f(x)$. We will now show that arithmetic operations with m known implicitly can be performed efficiently by representing all intermediary results $r \in \mathbb{Z}/(n)$ with polynomials $h(x) \in R$ of degree smaller than $\deg(f(x)) = t$ such that $h(m) \equiv r \pmod{n}$.

Given two polynomials $h(x), h'(x) \in R$ of degree smaller than t. Then polynomials representing the sum and product of $h(m)$ and $h'(m)$ can be found by adding respectively multiplying $h(x)$ and $h'(x)$ together and finally reducing the result modulo $f(x)$. A polynomial $r(x)$ representing the inverse of $h(m)$ can be found by using the extended Euclidean algorithm, i.e. by finding two polynomials $r(x)$ and $s(x)$ such that $r(x)h(x) + s(x)f(x) = \gcd(h(x), f(x))$. The inverse of

$h(x)$ is $r(x)$ if the gcd is 1. Otherwise, if $\gcd(h(x), f(x)) \neq 1$ then we have either found a nontrivial factor of n or $f(x)$. It is also possible to test equality, since $h(m) \equiv h'(m) \pmod{n}$ implies $\deg(\gcd(h(x) - h'(x), f(x))) \geq 1$. Either we have $\deg(\gcd(h(x) - h'(x), f(x))) = t$ and thus $h(m) \equiv h'(m) \pmod{n}$ or we have $\deg(\gcd(h(x) - h'(x), f(x))) = 1$ and $h(m) \not\equiv h'(m) \pmod{n}$ or we have found a nontrivial factor of $f(x)$. Hence we have shown that we can compute efficiently a polynomial $g'(x)$ of degree smaller than t such that

$$g(m) \equiv g'(m) \pmod{n}$$

or find a nontrivial factor of either n or $f(x)$.

A factor of n would mean that the secret key is found. If a factor of $f(x)$ is found then we can rerun the algorithm with $g(x)$ and each of the new factors of $f(x)$. Since the degree of $f(x)$ is t we will compute $g(x)$ in at most $2t$ rings $R_i = \mathbb{Z}[x]/(n, f_i(x))$ where $f_i(x)$ are factors of $f(x)$.

On the other hand if we find $g'(x)$ then we compute the gcd of $g'(x)$ and $f(x)$. From $g'(m) \equiv f(m) \equiv 0 \pmod{n}$ follows that $(x - m)$ is a divisor of the gcd. Thus when this gcd is in fact a linear polynomial then we can find m. We can now describe the algorithm as follows.

Algorithm 2. Given an RSA-modulus n with unknown factorization, a polynomial $f(x)$ of small degree and a rational function given by a short straight-line program (i.e. a short sequence of arithmetic operations to compute $g(x)$ from the set $\{x\} \cup \mathbb{Z}/(n)$). Then this algorithm tries to find a solution m to $f(m) \equiv g(m) \equiv 0 \pmod{n}$.

Step 1: Use square free factorization (e.g. [2, Algorithm 3.4.2]) to find

$$f(x) = \prod_{i=1}^{t} f_i(x)^i$$

where $f_i(x)$ are square free polynomials. If no factorization is found here (i.e. $f(x)$ is square free) then continue with step 2. Otherwise call this algorithm recursively with $n, f_i(x)$ and $g(x)$ for all $1 \leq i \leq t$ and return the union of solutions found.

Step 2: Let $R = \mathbb{Z}[x]/(n, f(x))$ and compute $g(x)$ over R.

If in any step a nontrivial factor of n is found then print this factor and terminate the algorithm.

If in any step a nontrivial factor $f'(x)$ of $f(x)$ is found then call this algorithm recursively with $n, f'(x), g(x)$ and with $n, f(x)/f'(x), g(x)$ and return the union of the solutions of this two calls.

Step 3: If no exception in Step 2 occurs then we get $g'(x) = g(x)$ over R where $g'(x)$ is a polynomials whose degree is smaller than $\deg(f(x))$. Now compute $r(x) = \gcd(f(x), g'(x))$.

If $r(x)$ is a constant then return 'no solution has been found'. If $r(x)$ is a linear polynomial then try to solve $r(m) \equiv 0 \pmod{n}$. This either finds a solution m or a nontrivial factor of n.

If $r(x)$ is a polynomial of degree larger than 1 then return that the algorithm is unable to solve $r(m) \equiv 0 \pmod{n}$.

Remark. In the situation, where we use this algorithm m will be the unique solution over $\mathbb{Z}/(n)$. This is not a sufficient condition since the algorithm can not find m when there is more than one solution to $g(m) = 0$ over $\mathbb{Z}[x]/(n, f(x))$. Therefore, our algorithm may sometimes fail. Fortunately, some of the attacks presented later in this paper allow a more rigorous analysis.

5 Partially known plaintext attack

In this section, we consider the security of the cryptosystem under the assumption that some part of the plaintext is known. We ask for the largest fraction of plaintext that can be recovered from the ciphertext when the rest of the plaintext is known. Hereby we assume an ideal situation for the attacker, i.e. we assume that the known bits are consecutive or simply those that help most.

Coppersmith has shown that $1/k$ of the bits of the plaintext can be recovered if a univariate equation of degree k over the plaintext is known. (See Theorem 1) This shows that up to $1/e$ unknown plaintext bits can be recovered from an RSA-encryption when the rest of the plaintext is known.

The attacks on KMOV in this section are based on the fact that the ciphertext (c_x, c_y) and the plaintext (m_x, m_y) are points on the same curve, i.e. that we can derive b from the ciphertext such that

$$m_x^3 + b \equiv m_y^2 \pmod{n}. \tag{2}$$

When the plaintext is partially known then we can eventually solve this equation. The multivariate version of Coppersmith's algorithm [3, Section 3]) can be applied when about 1/6 of the bits of the plaintext are unknown.

Here, we present another method that can tolerate up to 1/2 of unknown plaintext but that is less flexible since either m_x or m_y must be completely known.

Theorem 3. *Let n, e be a public key for KMOV and $C = (c_x, c_y)$ be the encryption of a message $M = (m_x, m_y)$. Then M can be computed efficiently given n, e, C and either m_x or m_y.*

Proof. Since (c_x, c_y) and (m_x, m_y) are points on the same elliptic curve we have

$$c_x^3 - c_y^2 \equiv m_x^3 - m_y^2 \pmod{n}. \tag{3}$$

When either m_x or m_y are known then Equation (3) becomes a univariate equation of degree 2 or 3 in the missing plaintext. Hence, we can apply the algorithm described in Section 4. Since there is no guarantee that the algorithm works in general we have to analyze this special case, which fortunately is simple enough to be analyzed rigorously.

Assume that m_x is known and m_y is unknown. Then we compute $e \cdot (m_x, y)$ over $\mathbb{Z}[y]/(y^2 - m_x^3 - b, n)$. It can be shown by induction over k and using the definition of the addition on elliptic curves that $k \cdot (m_x, y) \equiv (r_k, s_k y) \pmod{n}$ for two integers r_k, s_k. Thus we will finally get an equation $\phi(s_e y) \equiv c_y \pmod{n}$, which is solvable when $s_e \not\equiv 0 \pmod{n}$. If, however, $s_e \equiv 0 \pmod{n}$ then C is a point of order 2 and it follows from $M \equiv d \cdot C \pmod{n}$ that $C = M$. Hence, M is always computable.

Now assume that m_y is known and m_x is unknown. Then we will compute $e \cdot (x, m_y)$ over $\mathbb{Z}[x]/(x^3 + b - m_y^2, n)$. As before it can be shown by induction over k that $k \cdot (x, m_y) \equiv (r_k x, s_k) \pmod{n}$ for some integers r_k, s_k. Thus we finally have to solve the equation $\phi(r_e x) \equiv c_x \pmod{n}$, which is possible when $r_e \not\equiv 0 \pmod{n}$. Again we have to treat the case $r_e \equiv 0 \pmod{n}$ specially. It can be observed that $c_x \equiv 0 \pmod{n}$ and $a \equiv 0 \pmod{n}$ implies $2 \cdot C \equiv -C \pmod{n}$. Therefore C is a point of order 3 and hence M can be found easily. □

Example. Let $n = 493$ and $e = 7$ be the public key of KMOV. Assume that we know the ciphertext $C = (214, 358)$ and $m_y \equiv 229 \pmod{n}$, which is one half of the plaintext. First, we derive b from the ciphertext C and have

$$m_x^3 + b - m_y^2 \equiv m_x^3 + 297 \equiv 0 \pmod{493}.$$

Now we encrypt the point $P = (x, 229)$ over $\mathbb{Z}[x]/(x^3 + 297, 493)$ and get

$$C \equiv (12\,x, 358) \pmod{493}.$$

Therefore we have $m_x \equiv 214 \cdot 12^{-1} \equiv 100 \pmod{493}$.

6 Attacks in broadcast applications

In this section, we consider the situation of a broadcast application where a message is encrypted with different public keys and sent to the corresponding users. An attacker who intercepts some of these messages can sometimes combine the information he gained in such a way that he can learn the encrypted message. In particular, we will consider the following two situations:

1. All ciphertexts c_i are the encryption of the same message m.
2. The ciphertexts c_i are the encryption of linearly related messages

$$m_i \equiv \alpha_i m + \beta_i \pmod{n_i}$$

for some m where α_i and β_i are known constants.

We will review the security of RSA in broadcast applications before describing the attack against KMOV, since it is sometimes overlooked that Coppersmith has improved Håstad's result [6].

A simple method can be used when at least e RSA encryptions of the same message m encrypted with the same public exponent e are known, i.e. when the ciphertext c_i are known such that

$$c_i \equiv m^e \pmod{n_i} \text{ for } 1 \leq i \leq e.$$

From these equations we can derive C such that

$$C \equiv m^e \pmod{\prod_{i=1}^{e} n_i}.$$

Since $m^e < \prod_{i=1}^{e} n_i$ it follows that m can be found from C by computing the e-th root of C over $\mathbb{Z}$.

This simple method is no longer possible when we have k messages that are encrypted are not equal but linearly related. In this case, we will generally have a polynomial equation we can derive k equations

$$f_i(m) \equiv 0 \pmod{n_i} \text{ for } 1 \leq i \leq k.$$

We multiply the polynomials $f_i(m)$ by the inverse of their leading coefficient and possibly by a power of m such that the resulting polynomials are all monic polynomials of equal degree. Then we use the Chinese Remainder Theorem to derive an equation

$$F(m) \equiv 0 \pmod{N} \text{ where } N = \prod_{i=1}^{k} n_i. \tag{4}$$

The polynomial F is monic and thus we can use Coppersmith's algorithm if $|m| < N^{1/\deg(F)}$ to find m. This attack has apparently been described by Shimizu in [19].

A small improvement of this method is possible when the degrees of the polynomials $f_i(m)$ are different. Instead of multiplying them by a power of m it might be possible to compute powers of the polynomials itself. Since $f_i(m) \equiv 0 \pmod{n_i}$ implies $f_i(m)^{t_i} \equiv 0 \pmod{n_i^{t_i}}$ we can thus gain an equation $F(m) \equiv 0 \pmod{N}$ for a larger N. Given for example two RSA encryptions with $e_1 = 5$ and $e_2 = 3$ and a Rabin encryption of linearly related messages $m_i \equiv \alpha_i m + \beta_i \pmod{n_i}$ we can derive the following equations

$$\begin{aligned}(\alpha_1 m + \beta_1)^5 - c_1 &\equiv 0 \pmod{n_1}\\ (\alpha_2 m + \beta_2)^3 - c_2 &\equiv 0 \pmod{n_2}\\ (\alpha_3 m + \beta_3)^2 - c_3 &\equiv 0 \pmod{n_3}\end{aligned}$$

From these equations we compute

$$\begin{aligned}m\left((m + \beta_1\alpha_1^{-1})^5 - c_1\alpha^{-5}\right) &\equiv 0 \pmod{n_1}\\ \left((m + \beta_2\alpha_2^{-1})^3 - c_1\alpha^{-3}\right)^2 &\equiv 0 \pmod{{n_2}^2}\\ \left((m + \beta_3\alpha_3^{-1})^2 - c_1\alpha^{-2}\right)^3 &\equiv 0 \pmod{{n_3}^3}.\end{aligned}$$

All these equations are defined by monic polynomials of degree 6. Thus we can use the Chinese Remainder Theorem to get an equation

$$F(m) \equiv 0 \pmod{n_1 n_2^2 n_3^3}$$

where F is monic of degree 6 and $|m| < (n_1 n_2^2 n_3^3)^{1/6}$.

When KMOV is used then a message (m_x, m_y) can often be found when only 3 encryptions of the same message but with 3 different public keys are known. In particular, we have the following theorem.

Theorem 4. *Let $t \geq 1$, n_1, n_2, n_3 be the moduli of 3 different KMOV keys, $n = \max(n_1, n_2, n_3)$ and $\hat{n} = \min(n_1, n_2, n_3)$. Given the 3 ciphertexts of a randomly chosen message $M = (m_x, m_y) \in \{0, \ldots, \hat{n} - 1\}^2$ encrypted with these 3 keys then M can be found in time $O(t^2 \log(n)^3)$ with probability $1 - 1/t$.*

Proof. Because of Equation (1) we can derive b_i from the ciphertext such that

$$-b_i \equiv m_x^3 - m_y^2 \pmod{n_i} \text{ for } i \in \{1, 2, 3\}.$$

Thus we can find

$$b \equiv m_x^3 - m_y^2 \pmod{n_1 n_2 n_3}$$

for some $-\hat{n}^2 \leq b < n_1 n_2 n_3 - \hat{n}^2$. Moreover, since $m_x^3 - m_y^2$ must lie in the same interval it follows $b = m_x^3 - m_y^2$. We expect that m_y^2 is much smaller than m_x^3 and that therefore $m_x \approx b^{1/3}$, and we will show that m_x can be found with high probability by using this approximation.

Let $m_0 = \lceil b^{1/3} \rceil$. Then it is possible to find M in time $O(t^2 \log(n)^3)$ when $m_0 \leq m_x \leq m_0 + (4/3)t^2$. Indeed, we test for every $m_0 \leq m_x' \leq m_0 + (4/3)t^2$ whether the integer ${m_x'}^3 - b$ is a square. If this is the case then we let $m_y' = ({m_x'}^3 - b)^{1/2}$ and check whether the encryption of (m_x', m_y') with one of the public keys is equal to the corresponding ciphertext. This can be done in time $O(\log(n)^3)$ for every m_x'.

Thus it remains to show that the probability for $m_0 \leq m_x \leq m_0 + (4/3)t^2$ is at least $1 - 1/t$ for a randomly chosen message. Assume that $m_x \geq \hat{n}/t$ and let $\gamma = (4/3)t^2$. Then we have $m_y/t \leq m_x$ and thus $m_y^2 \leq (3/4)\gamma m_x^2 \leq \gamma((3/4)m_x^2 + (\gamma - (3/2)m_x)^2) = m_x^3 - (m_x - \gamma)^3$. Hence it follows $m_x^3 \geq m_x^3 - m_y^2 \geq (m_x - \gamma)^3$ and therefore that $m_0 \leq m_x \leq m_0 + (4/3)t^2$. Hence, when the attack fails we have $m_x < \hat{n}/t$ and the probability of this event is $1/t$. □

When the messages are not equal but linearly related then we can derive equations of the form $(\gamma_i m_y + \delta_i)^2 - (\alpha_i m_x + \beta_i)^3 \equiv b_i \pmod{n_i}$. Such equations can be combined with the Chinese Remainder Theorem to one equation of degree 3 in two unknowns m_x and m_y, i.e. from k messages we find f such that

$$f(m_x, m_y) \equiv 0 \pmod{\prod_{i=1}^{k} n_i}.$$

Applying Coppersmith's result we can hope for a solution if

$$\max(|m_x|, |m_y|) < \left(\prod_{i=1}^{k} n_i\right)^{(1/6-\epsilon)}$$

for some $\epsilon > 0$. This implies that the ciphertexts of 6 related messages might give enough information to recover the plaintext. This theoretical result should, however, be compared to our experimental results in section 9.

7 Attacks based on related messages for the same user

In this section, we discuss the situation where two related messages are both encrypted with the same public key. This situation has been analyzed by Coppersmith et al. for RSA [4]. They have shown that the ciphertext can be found from two encryptions with exponent e if it is computationally feasible to compute the gcd of two polynomials of degree e. They conclude that the attack is possible if the size of the public exponent e is smaller than about 32 bits.

The attack on KMOV presented here does not depend on the public parameter e. It is therefore not possible to prevent this attack by choosing e large. Let (m_x, m_y) and $(\tilde{m}_x, \tilde{m}_y)$ be two plaintexts that are related by known linear relations

$$\tilde{m}_x \equiv \alpha m_x + \gamma \tag{5}$$
$$\tilde{m}_y \equiv \beta m_y + \delta. \tag{6}$$

Assume that we know the encryption of these two messages, which is given by

$$(c_x, c_y) \equiv e \cdot (m_x, m_y) \pmod{n} \tag{7}$$
$$(\tilde{c}_x, \tilde{c}_y) \equiv e \cdot (\tilde{m}_x, \tilde{m}_y) \pmod{n} \tag{8}$$

From the ciphertext we can derive the curves $E_{0,b}(n)$ and $E_{0,\tilde{b}}(n)$ on which the points (m_x, m_y) and $(\tilde{m}_x, \tilde{m}_y)$ must lie. Thus we have

$$m_x^3 + b - m_y^2 \equiv 0 \pmod{n}$$
$$(\alpha m_x + \gamma)^3 + \tilde{b} - (\beta m_y + \delta)^2 \equiv 0 \pmod{n}$$

These two equations allow us to express m_y as a polynomial w in m_x. If we set

$$w(x) = \frac{(\alpha x + \gamma)^3 - \beta^2 x^3 - \delta^2 + \tilde{b} - \beta^2 b}{2\beta\delta}$$

then $w(m_x) \equiv m_y \pmod{n}$. Now let $f(x) = x^3 - w(x)^2 + b$, which is a polynomial of degree 6. From (1) follows $f(m_x) \equiv 0 \pmod{n}$. Next, we compute $e \cdot (x, w(x)) \equiv (h(x), j(x)) \pmod{n}$ over $\mathbb{Z}[x]/(n, f(x))$. Since we know the result of this encryption explicitly we have the equations

$$h(m_x) \equiv c_x \pmod{n} \tag{9}$$
$$j(m_x) \equiv c_y \pmod{n}. \tag{10}$$

Finally, we compute $\gcd(f(x), h(x) - c_x)$ and hope to find a linear polynomial of the form $\lambda(x - m_x)$, which allows us to find m_x.

Remark. The same attack would work even when the relation between (m_x, m_y) and $(\tilde{m}_x, \tilde{m}_y)$ is not linear but given by a polynomial relation whose degree is small.

When only one relation is known (e.g. between m_x and $\tilde{m}_x$ but not between m_y and $\tilde{m}_y$) then it is still possible to recover the plaintext when e is small. In

that case, we have to compute the gcd between two polynomials of degree e^2. And this seems possible if e is smaller than about 2^{16} [7]. If this method is successful it finds m_x, afterwards m_y can be found using the method of Section 5. Thus, it is sometimes possible to recover the plaintext of two related messages even if one of the two text blocks (m_x, m_y) is chosen randomly for every message.

8 Implication on related cryptosystems

Demytko's cryptosystem [5] uses, contrary to KMOV, only one coordinate to represent messages. This difference seems to be crucial, as the attacks presented in this paper can not be applied to Demytko's cryptosystem. Other proposed cryptosystems only vary the type of curve that is used for the encryption and use like KMOV both coordinates of a point to represent messages [10, 13]. Our attacks work in almost the same way against these cryptosystems too. For example Koyama uses in [10] singular cubic curves of the form

$$y^2 + axy \equiv x^3 \pmod{n}, \tag{11}$$

where the plaintext is a pair (m_x, m_y) and a is chosen such that the Equation (11) with $x = m_x$ and $y = m_y$ is satisfied. Koyama claimed that this cryptosystem is provably as secure as RSA, but faster than RSA. However, this claim hold only one day. Shamir presented at Eurocrypt'95 an attack which showed that one half of the plaintext can be found when the other half is known. Because of (11) the plaintext can also be recovered when at most 1/6 of m_x and m_y is unknown. When 6 linearly related messages are known then a Håstad attack is possible. The claim in [10] that the scheme is as secure as RSA in broadcast application is therefore not justified. The author wrongly assumes in the proof of section 5.2 of [10] that his elliptic curve cryptosystem cannot be weaker than RSA since he shows only that a successful attack on RSA would imply a successful attack on his scheme. The inversion of this implication is missing. Finally, we can perform a similar attack on Koyama's scheme as on KMOV when the ciphertexts of two related messages encrypted with the same public key are known. Again, this contradicts the conclusions the author draws from Theorem 4 as our attack shows that there is no reason to assume that Koyama's scheme is as secure as RSA.

9 Experimental results

This paper contains a few algorithms that have not been proven to work in all cases. We have therefore implemented all attacks in order to check their effectiveness. The attacks based on lattice basis reduction (i.e. based on [3]) are very computation expensive. This is specially the case when lattices of large dimension are involved. Therefore, we could not verify experimentally that all theoretical bounds are reachable. Fortunately, more knowledge can often help to reduce the dimension of the lattices.

A Håstad attack against KMOV with a 512-bit modulus and 6 linearly related messages seems to be computationally infeasible. But if 8 linearly related messages are known then a lattice basis of size 17 and 2400-digit integers has to be reduced. The package LiDIA, which implements a very sophisticated lattice basis reduction algorithm proposed by Schnorr and Euchner [18], can reduce such a lattice basis in about 2 weeks on an Ultra Sparc. The same attack with 9 linearly related messages can be done in about 15 minutes by reducing a lattice basis of size 6.

We observed that the attacks in Section 6 succeed almost always and we have not observed a failure in any of the other attacks.

10 Countermeasures

One possibility to avoid the attacks in this paper is to randomize some parts of the plaintext before the encryption. We propose that somewhat more than 1/5 of the bits in both coordinates of a point should be chosen randomly. This avoids the attacks presented in Section 5. Moreover, e should not be chosen too small, since a small e would give yet other small modular equations over the plaintext that can be combined with $m_x^3 + b \equiv m_y^2 \pmod{n}$ for even more effective attacks. Since the degree of the equations resulting from division polynomials (see e.g. [12]) is e^2 we suggest to choose e at least 16 bits long. These propositions require, of course, a careful analysis.

11 Conclusions

The attacks in this paper show that it is very dangerous when a cryptosystem leaks a modular relation of small degree on the message. Furthermore these attacks are an example for the fact that a more complex looking cryptosystem not necessarily is more secure than a simple looking one. The comparison of RSA and Koyama's scheme shows that a security analysis that considers only complete messages (i.e. showing that ability to decrypt all messages in one system implies that messages encrypted with the other system can also be decrypted) should not be used alone for comparing the security of two cryptosystems.

Acknowledgements

I'm grateful to Marc Joye for many comments on the paper. I would also like to thank Arjen Lenstra, Dan Boneh and Shai Halevi for answering questions about lattice basis reductions.

References

1. J. Borst. Public key cryptosystems using elliptic curves. Master's thesis, Eindhoven University of Technology, Feb. 1997.

2. H. Cohen. *A Course in Computational Algebraic Number Theory.* Number 138 in Graduate Texts in Mathematics. Springer Verlag, 1993.
3. D. Coppersmith. Finding a small root of a univariate modular equation. In *Advances in Cryptology - EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 155–165. Springer Verlag, 1996.
4. D. Coppersmith, M. Franklin, J. Patarin, and M. Reiter. Low exponent RSA with related messages. In *Advances in Cryptology - EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 1–9. Springer Verlag, 1996.
5. N. Demytko. A new elliptic curve based analogue of RSA. In T. Helleseth, editor, *Advances in Cryptology — EUROCRYPT '93*, volume 765 of *Lecture notes in computer science*, pages 40–49. Springer-Verlag, 1994.
6. J. Håstad. Solving simultaneous modular equations of low degree. *SIAM J. Computing*, 17(2):336–341, Apr. 1988.
7. M. Joye and J.-J. Quisquater. Overview and security analysis of RSA-type cryptosystems against various attacks. In *Proc. of DIMACS workshop on network threats*, Nov. 1996.
8. M. Joye and J.-J. Quisquater. Protocol failure for RSA-like functions using Lucas sequences and elliptic curves over a ring. In M. Lomas, editor, *Security Protocols*, volume 1189 of *Lecture Notes in Computer Science*, pages 93–100. Springer Verlag, 1997.
9. N. Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48(177):203–209, 1987.
10. K. Koyama. Fast RSA-type schemes based on singular cubic curves $y^2 + axy = x^3 \pmod{n}$. In *Advances in Cryptology - EUROCRYPT '95*, volume 921 of *Lecture Notes in Computer Science*, pages 329–340. Springer, 1995.
11. K. Koyama, U. Maurer, T. Okamoto, and S. Vanstone. New public-key schemes based on elliptic curves over the ring Z_n. In J. Feigenbaum, editor, *Advances in Cryptology - CRYPTO '91*, volume 576, pages 252–266. Springer Verlag, 1992. Lecture Notes in Computer Science.
12. K. Kurosawa, K. Okada, and S. Tsujii. Low exponent attack against elliptic curve RSA. In *Advances in Cryptology - ASIACRYPT 94*, volume 917, pages 376–383. Springer Verlag, 1995.
13. H. Kuwakado and K. Koyama. Efficient cryptosystems over elliptic curves based on a product of form-free primes. *IEICE Transactions on fundamentals of electronics, communications and computer sciences*, E77-A(8):1309–1318, Aug. 1994.
14. H. Kuwakado and K. Koyama. Security of RSA-type cryptosystems over elliptic curves against Håstad attack. *Electronic Letters*, 30(22):1843–1844, Oct. 1994.
15. A. Menezes, editor. *Elliptic Curve Public Key Cryptosystems.* Kluwer Academic Publishers, 1993.
16. V. S. Miller. Use of elliptic curves in cryptography. In H. C. Williams, editor, *Advances in Cryptology - CRYPTO '85*, volume 218 of *Lecture Notes in Computer Science*, pages 417–426. Springer, 1986.
17. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.
18. C. P. Schnorr and M. Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. In L. Budach, editor, *Proceedings of Fundamentals of Computation Theory (FCT '91)*, volume 529 of *Lecture Notes in Computer Science*, pages 68–85. Springer Verlag, Sept. 1991.
19. H. Shimizu. On the improvement of the Håstad bound. In *1996 IEICE Fall Conference*, volume A-162, 1996. (In Japanese).

20. T. Takagi and S. Naito. The multi-variable modular polynomial and its applications to cryptography. In *7th International Symposium on Algorithm and Computation, ISAAC'96*, volume 1178 of *Lecture Notes in Computer Science*, pages 386–396. Springer Verlag, 1996.