Cryptanalysis of Koyama Scheme

Sahdeo Padhye

School of Studies in Mathematics, Pt.Ravishankar Shukla University
Paipur (C.G.) (Email: sahadeo_mathrsu@yahoo.com)

(Received July 18, 2005; revised and accepted Aug. 14 & Sep. 10, 2005)

Abstract

In this paper we analyze the security of Koyama scheme based on the singular cubic curve for some well known attacks. We provide an efficient algorithm for linearly related plaintext attack and identify isomorphic attack on Koyama scheme. Some other attacks are also discussed in this paper.

Keywords: Koyama scheme, public key cryptosystem, RSA, singular cubic curve

1 Introduction

As variant of standard RSA [27] public key cryptosystem different type of cubic curve, three public key cryptosystem [13, 14, 17] were introduced. Those are called Koyama scheme. In these schemes, two plaintexts m_x, m_y are used to form a point $M = (m_x, m_y)$ on the singular cubic curve over Z_n, and the ciphertext is a point $C = e \times M$ on the same curve. There security was based on factoring problem. Later, Seng et al. [31] have shown that all three schemes are equivalent to each other by an isomorphism mapping and become insecure if a linear relation is known between two plaintexts. For this attack, attacker has to compute the greatest common divisor (GCD) of two polynomials both of degree e, where e is the encryption exponent. This attack was found less efficient because of its slow speed.

The object of this paper is to propose a new algorithm for linearly related plaintext attack on Koyama schemes [13, 14, 17]. Our algorithm is different and more efficient than Seng et al. [31] algorithm. In our proposed algorithm, the attacker has to compute the GCD of two polynomials of degree six and of degree e. Next, in this article, we identify isomorphic attack. From this, an attacker can forge signature of receiver B without knowing B's secret key. For this attack, a singular cubic curve is needed, isomorphic to the curve corresponding to the plaintext. Historically it was searched by Koyama for the KOMV [15] scheme which was based on nonsingular elliptic curve. Finally, we extend some other attacks on the RSA scheme to the Koyama schemes.

2 Singulaer Cubic Curve

First we discuss some basic facts about singular cubic curve over the finite field F_p and the ring Z_n where n is the product of two distinct odd primes greater then 3.

Consider the congruence equation

$$y^2 + axy = x^3 + bx^2 \bmod p \quad (a, b \in Z_p) \tag{1}$$

The set of all solutions $(x, y) \in Z_p \times Z_p$ to Equation (1) denoted by $C_p(a, b)$ is called singular cubic curve.

Let F_p be a finite field with p elements and ${F_p}^*$ be the multiplicative group of F_p. Clearly the order of ${F_p}^*$ denoted by $\sharp {F_p}^* = p - 1$.

A nonsingular part of singularcubic curve denoted by $C_p(a, b)$ is defined as the set of solutions $(x.y) \in F_p \times F_p$ to Equation (1) excluding a singular point $(0, 0)$, but including the point at infinity, denoted by $\bigcirc$.

It is well known that the same addition laws defined by the chord and tangent method in the case of elliptic curve still holds in the singular cubic curve [19, 28]. For any point $P \in C_p(a, b)$. For the sum $P + \bigcirc$, by definition, is equal to P, which is also equal to $\bigcirc + P$. For $P = (x_0, y_0)$, we define $-P$ the additive inverse of P as the point $(x_0, -y_0 - ax_0)$. The sum of $P + (-P)$ is defined to be $\bigcirc$. For $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ with $P_1 \neq P_2$ the sum $P_1 + P_2 = (x_3, y_3)$ is calculated as follows:

$$\begin{aligned} x_3 &= \gamma^2 + a\gamma - b - x_1 - x_2 \\ y_3 &= \gamma(x_1 - x_3) - y_1, \end{aligned}$$

where

$$\gamma = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1}, & \text{if } (x_1, y_1) \neq (x_2, y_2), \\ \frac{3x_1^2 + 2bx_1 - ay_1}{2y_1 + ax_1} & \text{if} (x_1, y_1) = (x_2, y_2). \end{cases}$$

The existence of such addition law makes $C_p(a, b)$ a finite abelian group. In fact, the group structure of $C_p(a, b)$ is well known [7, 19]. For any $k \in F_p$ the multiplication operation $''\times''$ is defined as bellow :

$$k \times (x, y) = \overbrace{(x, y) + (x, y) + (x, y) + ... + (x, y)}^{k \text{ times over } C_p(a,b)}$$

An isomorphism between $C_p(a,b)$ and ${F_p}^*$ is defined in [10, 14] for the curve $(y-\alpha x)(y-\beta x)=x^3$ over ${F_p}^*$, where $\alpha,\beta \in {F_p}^*$, which is equivalent to Equation (1) with $a=-\alpha-\beta mod\, p$ and $b=-\alpha\beta mod\, p$. When $b=0$ we can put $\alpha=0$ and $\beta=-a(\neq 0)$.

An isomorphism mapping from $C_p(a,0)$ to ${F_p}^*$ and inverse of that are given in the following theorems.

Theorem 1 *[19] The mapping $\omega : C_p(a,0) \to {F_p}^*$ defined by $\omega : \bigcirc \to 1$ and $(x,y) \to 1+\frac{ax}{y}=\frac{x^3}{y^2}$ is a group isomorphism.*

The group isomorphism mapping $\omega^{-1} : {F_p}^* \to C_p(a,0)$ is defined by

$$\omega^{-1} : 1 \to \bigcirc \quad \text{and} \quad v \to (\frac{a^2v}{(v-1)^2}, \frac{a^3v}{(v-1)^3})$$

Hence, with this isomorphism, the order of $C_p(a,0)$ is denoted by $\#C_p(a,0)=p-1$.

Let n be the product of two large primes p and q (>3). Let $Z_n=(1,2,3,....,n-1)$ and ${Z_n}^*$ be a multiplicative group of Z_n and consider the congruence equation

$$y^2+axy=x^3+bx^2 \text{ over } Z_n \text{ where } a,b \in Z_n. \qquad (2)$$

The nonsingular part of a singular cubic curve over Z_n denoted by $C_n(a,b)$, is defined, as the set of solutions $(x,y) \in Z_n \times Z_n$ to Equation (2) excluding a singular points which are either congruent to $(0,0) modulo\ p$ or congruent to $(0,0) modulo\ q$, but including a point at infinity $\bigcirc$. By Chinese Remainder Theorem, $C_n(a,b)$ is isomorphic as a group to $C_p(a,b) \times C_q(a,b)$.

Although the addition is not always defined, the probability of such a case is negligible small for large p and q. Since we are taking p and q very large, there fore the addition operation on $C_n(a,b)$ can be defined.

By using Theorem 1 and Chinese Remainder Theorem, the following theorem holds:

Theorem 2 *[7] For (x_1,y_1) and (x_i,y_i) satisfying $(x_i,y_i) = i \times (x_1,y_1)$ over $E_n(a,0)$, we have $1+\frac{ax_i}{y_i}=(1+\frac{ax_1}{y_1})^i (mod\, n)$, i.e. $\frac{x_i}{y_i}=(\frac{x_1}{y_1})^i (mod\, n)$.*

2.1 Division Polynomial in Singular Cubic Curve

The notion of division polynomial allows us to compute multiple of a point in terms of the first coordinate. The division polynomials on the singular cubic curve $C_n(o,b)$ is given as follows.

Definition 1 *The division polynomials $\Psi_m(x,y)$ for the singular cubic curve $C_n(o,b)$ are defined inductively by,*

$$\begin{aligned}
\Psi_1 &= 1,\\
\Psi_2 &= 2y,\\
\Psi_3 &= 3x^4+4bx^3,\\
\Psi_4 &= 4y(x^6-2bx^5),\\
\Psi_{2m-1} &= \Psi_{m-2}\Psi_m^3-\Psi_{m-1}\Psi_{m+1}^3, \quad \text{if } m \geq 2\\
2y\Psi_2 m &= \Psi_m(\Psi_{m+2}\psi_{m-1}^2-\Psi_{m-2}\Psi_{m-1}^2) \text{ if } m \geq 3
\end{aligned}$$

Theorem 3 *[31] Let $C_n(0,b)$ be a singular cubic curve defined over the ring Z/nZ. If $P=(x,y) \in C_n(0,b)$, then the first coordinate of $m \times P$ is given by*

$$x(m \times P)=\frac{x_{m^3}}{\Psi_m(x,y)^2}=\frac{x^m}{\phi_m(x,y)},$$

where $\Psi_m(x,y)$ is the m^{th} division polynomial for $C_n(0,b)$ and $\phi_m(x,y)$ is the polynomial defined by

$$\phi_m(x,y)=\frac{\Psi_m(x,y)^2}{x^{m^2-m}}.$$

3 RSA Type Schemes Based on Singular Cubic Curves

Three RSA type schemes based on singular cubic curve over Zn are proposed in the following subsections.

3.1 Scheme I [14]

This cryptosystem is based on the singular cubic curve of the form

$$C_n(0,b) := y^2 \equiv x^3+bx^2 (mod\ n) \qquad (3)$$

where $n=pq$ is the product of two large primes. The encryption key e is chosen such that $(e,N)=1$ where $N=lcm(p-1,p+1,q-1,q+1)$.The decryption key d is chosen such that $ed \equiv 1 mod\ N$. The public key is the pair (n,e) and the private keys are d, p and q. To encrypt a plaintext pair $M=(m_x,m_y)$, the sender first computes $b=\frac{m_y^2-m_x^3}{m_x^2} (\text{mod } n)$ and then the ciphertext is computed as $C=e \times M$ on the singular cubic curve $C_n(0,b)$. The complete ciphertext is (C,b). The Receiver, who knows the decryption key d can get the plaintext (m_x,m_y) by computing $d \times (c_x,c_y)=(m_x,m_y)$ over the singular cubic curve $C_n(0,b)$.

3.2 Scheme II [17]

This cryptosystem is based on the singular cubic curve of the form

$$C_n(a,0) := y^2+axy \equiv x^3 (mod\ n)$$

where $n=pq$ is the product of two large primes. The encryption key e is chosen such that $(e,N)=1$ where $N=lcm(p-1,q-1)$. The decryption key d is chosen such that $ed \equiv 1 mod\ N$. The public key is the pair (n,e) and the private keys are d, p and q. To encrypt a plaintext pair $M=(m_x,m_y)$, the sender first computes $a=\frac{m_x^3-m_y^2}{m_x m_y} (\text{mod } n)$ and then the ciphertext is computed as $C=e \times M$ on the singular cubic curve $C_n(a,0)$.The complete ciphertext is (C,a). The Receiver, who knows the decryption key d can get the plaintext (m_x,m_y) by computing $d \times (c_x,c_y)=(m_x,m_y)$ over the singular cubic curve $C_n(a,0)$.

3.3 Scheme III [13]

This cryptosystem is based on the singular cubic curve of the form

$$C_n(a,b) := (y-\alpha x)(y-\beta x) \equiv x^3 (mod\ n)$$

where $n = pq$ is the product of two large primes. The encryption key e is chosen such that $(e, N) = 1$ where $N = lcm(p-1, q-1)$. The decryption key d is chosen such that $ed \equiv 1 mod\ N$. The public key is the pair (n, e) and the private keys are d, p and q. To encrypt a plaintext pair $M = (m_x, m_y)$, sender first chooses a randomly and computes . Then the ciphertext is computed as $C = e \times M$ on the singular cubic curve $C_n(\alpha, \beta)$.The complete ciphertext is (C, α, β). The Receiver, who knows the decryption key d can get the plaintext (m_x, m_y) by computing $d \times (c_x, c_y) = (m_x, m_y)$ over the singular cubic curve $C_n(\alpha, \beta)$.

Seng et al. [31] have given following two equivalence relations between Schemes I, II and III.

1) Reduction of Scheme II to Scheme I: The transformation $(x, y) \rightarrow (x, y + \frac{a}{2}x)$ will transform the curve $C_n(a, 0)$ to the curve $C_n(0, b)$ with $b = a^2 4$. Using this transformation one can reduce scheme II to Scheme I.

2) Reduction of Scheme III to Scheme I: The transformation $(x, y) \rightarrow (x, y - \frac{\alpha-\beta}{2}x)$will transform the curve $C_n(\alpha, \beta)$ to the curve $C_n(0, b)$ with $b = (\frac{\alpha-\beta}{2})^2$. Using this transformation, one can reduce Scheme III to the Scheme I.

4 An Efficient Algorithm for Linearly Related Plaintext Attack

Let us discuss the situation when two linearly related messages are both encrypted with the same public key. In 1995, Franklin and Reiter [6] identified such type of attack against RSA with public exponent 3. Later, it was extended for the exponent up to ≈ 32 bits by Patarin [21]. Further, it was generalized by Joye and Quisquater [10] to the other RSA type cryptosystems, like elliptic curve RSA. This attack was also generalized for any known polynomial relation between two messages and to any number of messages [3]. In an important paper Seng et al. [31] analyzed linearly related plaintext attack for the Koyama schemes [13, 14, 17]. For this attack, attacker has to compute greatest common divisor of two polynomials corresponding to those two plaintexts. Incidentally, the computation of greatest common divisor of two polynomials becomes less efficient when the encryption exponent e is considered quite large. We therefore reexamined linearly related plaintext attack on Koyama schemes and propose a different algorithm to deal with this situation. Our algorithm is more efficient with comparison to Seng et al. [31]. During our discussion we however confine to scheme-I, because scheme-II and scheme-III are reducible to the scheme-I.Let us we first recall the Seng et al. attack [31].

Seng et al. Attack [31].
Let $M_1 = (m_{x_1}, m_{y_1})$, and $M_2 = (m_{x_1} + \nabla, m_{y_1} + \nabla)$ be two linearly related plaintexts, and $C_1 = e \times M_1 = (c_{x_1}, c_{y_1})$, $C_2 = e \times M_2 = (c_{x_2}, c_{y_2})$ be the ciphertexts corresponding to plaintexts M_1 and M_2 respectively. Here ∇ is a known constant. By Theorem 3, we have

$$\begin{aligned} m_x^e - c_{1.x}\phi_e(m_x, .) &\equiv 0 \bmod n, \text{ and} \\ (m_x + \nabla)^e - c_{2.x}\phi'_e(m_x + \nabla, .) &\equiv 0 \bmod n, \end{aligned}$$

where ϕ_m is defined by

$$\phi_m(x, y) = \frac{\Psi_m(x, y)^2}{x^{m^2-m}},$$

for the curve $C_n(0, b)$ on which M_1 and C_1 lie. The function ϕ' is defined analogously for the curve on which M_2 and C_2 lie. Using the above relation the Seng et al. attack works as follows.

1) Let $F(x)$ and $G(x)$ be the polynomials over the ring Z/nZ, defined by

$$\begin{aligned} F(x) &= x^e - c_{1,x}\phi_{e,}(x, .), \\ G(x) &= (x + \nabla)^e - c_{2,x}\phi_{e,}(x + \nabla, .). \end{aligned}$$

2) We compute $H(x) = gcd(F(x), G(x))$, the gcd of $F(x)$ and $G(x)$ over the ring Z/nZ, which is with a very high probability, a polynomial of degree 1. Solving the polynomial $H(x)$ in x will give the value of m_x.

Proposed Algorithm For Linearly Related Plaintext Attack.
Now we propose our algorithm as follow. Let $M = (m_x, m_y)$ and $M' = (m'_x, m'_y)$ be two plaintexts linearly related by the known relations

$$\begin{aligned} m'_x &\equiv \alpha m_x + \gamma \\ m'_y &\equiv \beta m_x + \delta, \end{aligned}$$

where α, γ, β and δ are integers in $Z_n{}^*$. Assume that the encryption of the plaintexts (m_x, m_y) and (m'_x, m'_y) are given by

$$\begin{aligned} (c_x, c_y) &\equiv e \times (m_x, m_y) (\bmod n) \\ (c'_x, c'_y) &\equiv e \times (m'_x, m'_y) (\bmod n). \end{aligned}$$

From the above ciphertext we can derive the curves $C_n(0, b)$ and $C_n(0, b')$ upon which the plaintexts must lie. Thus we have

$$\begin{aligned} m_x^3 + bm_x^2 - m_y^2 &\equiv 0 (\bmod n) \\ (\alpha m_x + \gamma)^3 + b'(\alpha m_x + \gamma)^2 - (\beta m_y + \delta)^2 &\equiv 0 (\bmod n). \end{aligned}$$

By above two equations we can write m_y as a polynomial w in m_x with

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

By Equation (3) it is clear that $w(m_x) \equiv m_y(mod\ n)$. Now let $f(x) \equiv x^3 + bx^2 - w(x)^2(\mathrm{mod} n)$, which is a polynomial of degree 6. From Equation (3) we see that $f(m_x) \equiv 0(mod\ n)$ on $Z[x]/(n, f(x))$. Next we compute $e \times (x, w(x)) \equiv (h(x), j(x))(mod\ n)$ over $Z[x]/(n, f(x))$ using the division polynomial. Then we have the following equations

$$\begin{aligned} h(m_x) &\equiv c_x(mod\ n) \\ j(m_x) &\equiv c_y(mod\ n). \end{aligned}$$

Further, we compute $gcd(h(x) - c_x, f(x))$ which is a linear polynomial of the form $k(x - m_x)$. This gives us the plaintext m_x. Knowing this half of the plaintext $(m_x, m_y) = M$, we can compute the other half m_y by $w(m_x) = m_y$. Finally, because of the linear relation between M and M' we can compute the plaintext M'.

4.1 Comparison between Seng et al. Algorithm (SA) and Proposed Algorithm (PA)

SA- Let two linearly related messages are (x, y) and $(x + \Delta, y + \Delta)$.

1) In SA, attacker has to compute first coordinate of $e \times (x, y)$ and $e \times (x + \Delta, y + \Delta)$ by using the division polynomial. Let it be c_{1x} and c_{2x}.

2) In SA, the attacker has to compute the *gcd* of two polynomials $F(x)$ and $G(x)$ both of degree e, where $F(x) = x^e - c_{1x}\phi x, .)modn$, and $H(x) = (x + \Delta)^e - c_{2x}\phi e(x + \Delta, .)modn$.

3) SA depends up on e, hence, for higher values of e attack is less applicable.

PA- Let two linearly related messages are $M = (m_x, m_y)$ and $M' = (m'_x, m'_y)$, where $m'_x = \alpha m_x + \gamma$ and $m'_y = \beta m_x + \delta$.

1) In PA, attacker has to compute first coordinates of $e \times (x, w(x))$ where

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

Let $e \times (x, w(x)) = (h(x), J(x))$ and $f(x) \equiv x^3 + bx^2 - w(x)^2(mod\ n)$.

2) In PA, attacker has to compute the *gcd* of two polynomials $F(x) = h(x) - c_x$ and $f(x)$. Here, $F(x)$ is a polynomial of degree e and $f(x)$ is a polynomial of degree at most 6 (in particular if $\alpha = \beta = 1$ then $f(x)$ is a polynomial of degree 4).

3) PA does not depend upon the encryption exponent; hence, it is applicable for each value of e.

4) In PA, since one polynomial is of degree at most 6 so we may assume that the computational efficiency is faster than SA.

We compare both attacks by the following example.

Example 1 *Suppose we set,*
Keys: $p = 1237, q = 5683, e = 11,$
Plaintext: $M_1 = (54321, 67890), M_2 = (54411, 67980),$ *i.e.* $\gamma = \delta = \Delta = 90.\ \alpha = \beta = 1$
Ciphertext: $C_1 = (2687388, 3712394),$
$C_2 = (2387261, 3231021).$
Clearly, $b = 1793115$ *and* $b' = 4717526$

In Seng et al. attack,
$F(x) = x^{11} + 5229989x^{10} + 3440216x^9 + 1918724x^8 + 33716x^7 + 4214133x^6 + 528492x^5 + 65705x^4 + 5018141x^3 + 2203074x^2 + 3786039x + 3314999$
$G(x) = x^{11} + 6396991x^{10} + 4606503x^9 + 679657x^8 + 6778159x^7 + 6520626x^6 + 6319754x^5 + 806279x^4 + 3985603x^3 + 4360013x^2 + 435444x + 1937673.$
Then $gcd(F(x), G(x)) = 6975550 + x$. *Solving this, we gate* $x = 54321$.

In our proposed attack,
$F(x) = x^{11} + 5229989x^{10} + 3440216x^9 + 1918724x^8 + 33716x^7 + 4214133x^6 + 528492x^5 + 65705x^4 + 5018141x^3 + 2203074x^2 + 3786039x + 3314999$
$G(x) = 5688501x^4 + 243866x^3 + 2935716x^2 + 1708749x + 1353731$
Then $gcd(F(x), G(x)) = 6975550 + x$. *Solving this, we gate* $x = 54321$.

The attack proposed by Seng et al. [31] is similar to that proposed in [10], but in [10] $F(x)$ and $G(x)$ are polynomials of degree e^2, where as in Seng et al. attack [31] involves only polynomial of degree e and is there fore more efficient than [10]. In our algorithm , $h(x) - c_x$ is same as $F(x)$ in the Seng et al. attack (by the assumption of $w(m_x) = m_y$), but we replace $G(x)$ by a polynomial of degree at most 6. So we conclude that our proposed algorithm is more efficient than Seng et al. [31] attack for higher values of e (i.e. for $e \geq 5$). In addition, our proposed attack does not depend up on e so it is applicable for any value of e.

5 Isomorphic Attack

The idea behind the isomorphic attack is based on the isomorphic property of two singular cubic curves. Such type of attack was first time identified by Koyama for the KMOV scheme [15]. We first give definition and the isomorphic property as follows.

Definition 2 *Let $n = pq$ (p, q are primes), and $C_n(0, b_1)$ and $C_n(0, b_2)$ be singular cubic curves such that*

$$C_n(0, b_1) : y^2 = x^3 + b_1x^2 (mod\ n),$$
$$C_n(0, b_2) : y^2 = x^3 + b_2x^2 (mod\ n).$$

$C_n(0, b_1)$ and $C_n(0, b_2)$ are isomorphic if there exist $u_p \in Z_p^*$ and $u_q \in Z_q^*$ such that

$$b_2 \equiv u_p^2 b_1 (mod\ p), \text{ and } b_2 \equiv u_q^2 b_1 (mod\ q).$$

By using the property of singular elliptic curve over field and Chinese Remainder Theorem, the following isomorphic property of singular cubic curve over ring is shown [28] as bellow:

Let $C_n(0, b_1) : y^2 = x^3 + b_1x^2 (mod\ n)$ and $C_n(0, b_2) : y^2 = x^3 + b_2x^2 (mod\ n)$ be two singular cubic curves. Let $M_1 = (m_{1x}, m_{1y})$, $C_1 = (c_{1x}, c_{1y}) \in C_n(0, b_1)$ and $M_2 = (m_{2x}, m_{2y})$, $C_2 = (c_{2x}, c_{2y}) \in C_n(0, b_2)$, where $C_1 = e \times M_1$ over $C_n(0, b_1)$ and $C_2 = e \times M_2$ over $C_n(0, b_2)$. Then the following statements are equivalent,

1) $C_n(0, b_1)$ and $C_n(0, b_2)$ are isomorphic
2) $b_2 \equiv u^2 b_1 (mod\ n)$ for some $u \in Z_n^*$
3) $c_{2x} \equiv u^2 c_{1x} (mod\ n)$, $c_{2y} \equiv u^3 c_{1y} (mod\ n)$ for some $u \in Z_n^*$
4) $m_{2x} \equiv u^2 m_{1x} (mod\ n)$, $m_{2y} \equiv u^3 m_{1y} (mod\ n)$ for some $u \in Z_n^*$

If C_1, C_2 and M_1 satisfying the above Item 3) are given, then M_2 can be easily obtained by computing Item 4). It is not difficult to check whether or not Item 3) holds.

Suppose, an attacker A wants to victimize B by forge signature on a plaintext $M = (m_x, m_y)$ without B's consent. For this, A generates another message M' with B's public key n_B and random integer u:

$$M' = (u^2 m_x (\mathrm{mod} n_B), u^3 m_y (mod\ n_B)).$$

And sends M' to B. B makes a signature $S' = (s'_x, s'_y)$ for M' with his secret key d_B:

$$S' = d_B \times M' \text{ over } C_{n_B}(0, b'_B).$$

Then, A computes the signature $S = (s_x, s_y) = (u^{-2} s'_x (mod\ n_B), u^{-3} s'_y (\mathrm{mod} n_B))$. Which is B's signature for the message M.

Note that the curve $C_{n_B}(0, b_B)$ contains points (M, S) and the curve $C_{n_B}(0, b'_B)$ contains points (M', S').

Using this technique A can forge B's signatures without B's secret key.

6 Homomorphic Attack

Originated from homomorphic property (i.e. $k \times [P+Q] = k \times [P] + k \times [Q]$), the known homomorphic attacks such as common modulus attack [30], chosen message attack [32], garbage man-in-the -middle attack [9, 11] etc. are also found admissible to Koyama schemes [13, 14, 17]. We name Carol to the attacker, Alice to the receiver (or message signer) and Bob to the sender to discuss these attacks as follows.

6.1 Common Modulo Attack

In 1983, Simmons [30] pointed out that the use of common modules in the RSA cryptosystem is dangerous. Indeed, if a message M is sent to two users who have coprime public encryption keys (e_1 and e_2 say), then the message M can be recovered. Suppose the ciphertext corresponding to the plaintext M are $C_1 = M_1^{e_1} mod\ n$ and $C_2 = M_2^{e_2} mod\ n$, then by extended Euclidean algorithm [16], Carol can compute u and v such that $ue_1 + ve_2 = 1$, and he can easily get the indented plaintext M by computing $C_1^u C_2^v mod\ n = M^{e_1 u + e_2 v} mod\ n = M$. This attack is called common modulus attack. Joye et al. [11] have shown that common modulus attack is applicable to KMOV [15] scheme. Also, Demytko scheme [5] is vulnerable to said attack [2]. Below, we show this attack is applicable to Koyama scheme.

[Input] Two ciphertexts $C_1 = (c_{x_1}, c_{y_1}), C_2 = (c_{x_2}, c_{y_2})$, common modulus n, and encryption keys e_1, e_2.

[Step 2] By using extended Euclidean algorithm Carol computes u and v such that $eu + kv = 1$.

[Step 3] Carol computes

$$\begin{aligned} & u \times (c_{x_1}, c_{y_1}) + v \times (c_{x_2}, c_{y_2}) \\ = \ & (u \times e_1 + v \times e_2) \times (m_x, m_y) \\ = \ & (m_x, m_y). \end{aligned}$$

[Output] The intended plaintext pair (m_x, m_y).

6.2 Chosen Message Attack

In a paper, Simmons and Norris [32] have shown that the RSA cryptosystem is not secure against chosen message attack. Using the chosen message attack, Carol can get the signature of Alice on any chosen message M. Also, the chosen message attack is applicable to all RSA type cryptosystem [11] based on elliptic curve and Lucas sequence that possesses homomorphic property. Below, we show that this attack is applicable to the Koyama schemes [13, 14, 17].

Suppose an attacker Carol wants to get the signature of a person Alice on the message pair (m_x, m_y). Then he proceeds as follows.

[Input] A message pair (m_x, m_y) and the key n, e of plaintext.

[Step 1] First Carol chooses k relatively prime to e and compute u and v such that $eu + kv = 1$.

[Step 2] Carol computes $M' = k \times (m_x, m_y) = (m'_x, m'_y)$.

[Step 3] Next, she ask Alice to sign on the document $M' = (m'_x, m'_y)$ and gets the signature $S' = d \times (m'_x, m'_y) = (s'_x, s'_y)$.

[Step 4] Consequently Carol can compute the signature S of M by

$$\begin{aligned} S &= u \times (s'_x, s'_y) + v \times (m_x, m_y) \\ &= u \times d \times k \times (m_x, m_y) + v \times (m_x, m_y) \\ &= d \times (u \times k + e \times v) \times (m_x, m_y) \\ &= (s_x, s_y). \end{aligned}$$

[Output] The signature S of M.

6.3 Garbage Man-in-the-middle Attack

If an attacker wants to get the intended plaintext M from a given ciphertext $C = (c_x, c_y)$ in the Koyama scheme, then by using the Garbage man-in-middle attack [9] he can get the intended plaintext as follows.

[Input] A message pair (c_x, c_y), n, e.

[Step 1] First Carol inspects $C = e \times (m_x, m_y) = (c_x, c_y)$.

[Step 2] $C' = (c'_x, c'_y) = k \times (c_x, c_y) = k \times e \times (m_x, m_y)$ for any chosen k relatively prime to e.

[Step 3] By Extended Euclidean Algorithm, Carol can computes $u, v \in Z_n$ such that $k \times u + e \times v = 1$.

[Step 4] She ask Alice to sign on (c'_x, c'_y) and get

$$\begin{aligned} S' &= (s'_x, s'_y) \\ &= d \times (c'_x, c'_y) \\ &= d \times k \times e \times (m_x, m_y) \\ &= k \times (m_x, m_y) \end{aligned}$$

[Step 5] Now, Carol can compute the original Message as

$$\begin{aligned} & u \times (s'_x, s'_y) + v \times (c_x, c_y) \\ = \; & u \times k \times (m_x, m_y) + e \times v \times (m - x, m_y) \\ = \; & (m_x, m_y) \end{aligned}$$

[Output] The intended plaintext.

7 Factoring Attack

Some other Attacks direct related to the factoring are also admissible to the Koyama schemes. If the secret primes factor p and q of RSA-modulus are improper chosen, then by factoring attacks [22, 34] one can recover the secret keys. Note that p and q must be generated [24] carefully to prevent the scheme form some well known factoring attacks. Some other factoring attacks were proposed by Silverman and Rivest [26, 29]. Furthermore, if a portion of bits of p or q is known, then the RSA moduli can be factorized [4, 25] all such type of attacks are automatically applicable to any cryptosystem based on factoring.

8 Partially Known Plaintext Attack

In his paper Koyama [17] conjectured that if an attacker knows one ordinate on the plaintext pair (m_x, m_y) corresponding to the ciphertext (c_x, c_y) then there is no problem to the security point of view in other words partially known plaintext attack is not applicable. In a paper [20] it has shown that the conjecture made by Koyama regarding the said attack is failed. Following theorem proved the partially known plaintext attack on Koyama schemes.

Theorem 4 *[20] For any (x_1, y_1) in the singular cubic curve $C_n(0, b)$. Let x_1 is known. Then for any k in Z_n, $k \times (x_1, y_1) \equiv (u_k, v_k y_1)$ over $Z[y]/(y^2 - x_1^3 - bx_1^2$, where, u_k, and v_k are two positive integers given by: $u_1 = x_1$, $v_1 = 1$, and*

$$u_k = \begin{cases} \frac{(3u_{\frac{k}{2}}^2 + 2bu_{\frac{k}{2}})^2}{4(u_{\frac{k}{2}}^3 + bu_{\frac{k}{2}}^2)} - b - 2u_{\frac{k}{2}}, & \text{if } k \text{ is even} \\ \frac{(u_1^3 + bu_1^2)(1 - v_{k-1})^2}{(u_1 - u_{k-1})^2} - b - u_1 - u_{k-1}, & \text{if } k \text{ is odd} \end{cases}$$

and

$$v_k = \begin{cases} v_{\frac{k}{2}} \{ \frac{3u_{\frac{k}{2}}^2 + 2bu_{\frac{k}{2}}}{2(u_{\frac{k}{2}}^3 + bu_{\frac{k}{2}}^2)} (u_{\frac{k}{2}} - u_k) - 1 \} \quad or \\ \quad \frac{(3u_{\frac{k}{2}}^2 + 2bu_{\frac{k}{2}})}{2(u_1^3 + bu_1^2) v_{\frac{k}{2}}} (u_{\frac{k}{2}} - u_k) - v_{\frac{k}{2}}, & \text{if } k \text{ is even} \\ \frac{1 - v_{k-1}}{u_1 - u_{k-1}} (u_1 - u_k) - 1, & \text{if } k \text{ is odd} \end{cases}$$

9 Some Other Attacks

9.1 Wiener's Attack

Wiener [33] has shown that if the secret key d is chosen too small, then it can be recovered. In his paper, Wiener proved the following theorem.

Theorem 5 *[33] Let $n = p.q$, with $q < p < 2q$, Let $d < \frac{1}{3} n^{\frac{1}{4}}$. Given (n, e) with $ed \equiv 1 mod\ \phi(n)$. One can efficiently recover d.*

Since, the prove of the above theorem does not depend on the encryption procedure, so it is also applicable to the Koyama schemes, as in the Koyama schemes [13, 14, 17] we have $ed \equiv 1 mod\ (lcm(p-1, q-1))$. So, Wiener attack is also applicable to Koyama schemes [13, 14, 17].

9.2 Lenstra Attack

In the year 1996 Boneh et al. identified a new attack against RSA when performed with Chinese remaindering [23]. Thereafter and independently, Lenstra [18] wrote a short memo on such type of attack which we call as Lenstra attack. In case of computational error, Boneh et

al. [1] showed how to recover the secret factors p and q of the public modulus n from two signatures of the same messages; a correct one and a faulty one. However, Lenstra [18] showed that faulty signature is required only. The attack proposed by Lenstra is as follows,

Let p and q be two primes and let $n = pq$. To sign upon any message m by using the Chinese remaindering, the signer proceeds as follows;

Signer first computes $s_p \equiv m_p^{d_p} mod\ p, s_q \equiv m_q^{d_q} mod\ q$, where $m_p \equiv m(mod\ p)$, $m_q \equiv m(mod\ q)$, $d_p \equiv d(mod\ p-1)$ and $d_q \equiv d(mod\ q-1)$.

Suppose an error is occurring at the time of the computation of s_p (let the faulty value of s_p is $\hat{s_p}$) and not during the computation of s_q. Now applying the Chinese Remaindering Theorem on $\hat{s_p}$ and s_q , let $\hat{s}$ be the faulty signature of m. Then gives the secret parameter q, where e is the public key.

Joye et al. [8] showed that the Lenstra attack is of very general nature and applies on all Chinese remaindering based cryptosystems. Joye et al. [8] generalized the Lenstra in a preposition as follows

Proposition 1 *[8] Let primes p and q whose product is n. Suppose that $s = S(m)$ is the signature of a message m and that is the faulty signature. $S : m \rightarrow S(m)$ is a RSA type signature function. If $\hat{s} \neq s\ mod\ p$ but $\hat{s} = s\ mod\ q$, then $gcd((S^{-1}(\hat{s}) - m) mod\ n, n)$ will give the secret factor q.*

Using the above preposition, Joye et al. [8] applied Lenstra attack to KMOV scheme [15] and Demytko [5].

Lenstra attack on Koyama scheme.
The said above preposition is applicable to Koyama scheme [17] as follows.

Suppose $(m_x, m_y) = M$ be the intended message to be signed. Let, $s = (s_x, s_y) = d \times (m_x, m_y)$ be the Koyama signature of message M. Suppose that the computation of the x-coordinate of $s = (s_x, s_y)$ is faulty. More precisely, if $\hat{s_x} \neq s_x(mod\ p)$ and $= \hat{s_x} = s(mod\ q)$, then applying the same argument as in the case of KMOV scheme given by Joye et al. [8], q can be recovered by computing $gcd(\frac{\hat{s}_x^{e^2}}{\Psi_e(\hat{s_x}, s_y)^2} - m\ mod\ n), n$(first term is computed by using the division polynomial for singular cubic curve [31]).

Acknowledgements

This work is supported under CSIR (JRF) scheme, India (2002).

References

[1] D. Boneh, R. A. DeMillllo, and R. J. Lipton, "On the importance of checking cryptographic protocols for faults," in *Eurocrypt'97*, LNCS 1233, pp. 37–51, Springer-Verlag, 1997.

[2] D. Bleichenbacher, M. Joye, and J. J. Quisquater, "A new and optimal chosen-message attack on RSA-type cryptosystems," in *Information and Communications Security*, LNCS 1233, pp. 302–313, Springer-Verlag, 1997.

[3] D. Coppersmith, M. Franklin, J. Patarin and M. Reiter, "Low exponent attack RSA with related messages ," in *Proceeding in EUROCRYPT'96*, LNCS 765, pp. 40–49, Springer-Verlag, 1996.

[4] D. Coppersmith, "Finding a small root of a bivariate integer equation;factoring with high bits known," in *Advances in Cryptology-Eurocrypt'96*, LNCS 1070, pp. 178–189, Springer-Verlag, 1996.

[5] M. Demytko, "A new elliptic curve based analogue to RSA," in *EUROCRYPT'93*, pp. 40–49, 1994.

[6] M. K. Franklin and M. K. Reiter, "A linear protocal failure for RSA with exponent three," in *Preliminary notes for Crypto'95 rump session.*

[7] D. Husemaller, *Elliptic Curves*, Springer Verlag. 1987.

[8] M. Joye, A. K. Lenstra and J. J. Quisquater, "Chinese remaindering in the presence of faults," *Journal of Cryptology*, vol. 12, no. 4, pp. 241–245, 1999.

[9] M. Joye, *Security Analysis of RSA-Type Cryptosystems*, Ph.D. Thesis UCL Cryptogroup, Universite Catholique De Louvain, Place Du Levant 3, B-1348 Louvain-La-Neuve, Belgium.

[10] M. Joye and J. J. Quisquater, "Protocol failure for RSA like functions using Lucas sequence and elliptic curves," in *Security Protocol*, LNCS 1189, pp. 93–100, Springer-Verlag, 1997.

[11] M. Joye, and J. J. Quisquater, "Cryptanalysis of RSA-type cryptosystem: A visit," *Theoretical Computer Science*, vol. 38, pp. 21–31, 1998.

[12] K. Koyama and H. Kuwakado, "Efficient cryptosystems over elliptic curves based on a product of form-free primes," *IEICE Transactions on Fundamentals*, vol. E77-A, pp. 1309–1318, 1994.

[13] K. Koyama and H. Kuwakado, "A new RSA-type scheme based on singular cubic curves $(y - \alpha x)(y - \beta x) \equiv x^3 (mod\ n)$," *IEICE Transactions on Fundamentals*, vol. E79-A, pp. 49–53, 1996.

[14] H. Kuwakado, K. Koyama, and Y. Tsuruoka, "A new RSA-type scheme based on singular cubic curves $y^2 \equiv x^3 + bx^2 (mod\ n)$," *IEICE Transactions on Fundamentals*, vol. E78-A, pp. 27–33, 1995.

[15] K. Koyama, U. M. Maurer, T. Okamoto and S. A. Vanstone, "New public key schemes based on elliptic curves over the ring," in *Crypto 91*, pp. 252–266, 1991.

[16] D. E. Knuth, *The Art of Computer of Computer Programming: Seminumerical Algorithms (Vol. 2)*, 2nd, Addison-Wesley, 1981.

[17] K. Koyama, "Fast RSA type scheme based on singular cubic curve $y^2 + axy = x^3 (mod\ n)$," in *Eurocrypt'95*, LNCS 921, pp. 329–339, Springer-Verlag, 1995.

[18] A. K. Lenstra , *Memo on the RSA Signature Generation in the Presence of Faults*, Sep. 1996.

[19] A. Menezes, *Elliptic Curve Public Key Cryptosystem*, Kluwer Acadamic, 1993.
[20] S. Padhye, "Partial known plaintext attack on Koyama scheme," *Information Processing Letters*, 2005 (To appear).
[21] J. Patarin, "Some serious protocal failure for with exponent e if less 32 bits," in *Presented at the Conference for Cryptography*, CIRM Luminy, France, Sep. 1995.
[22] J. M. Pollard, "Theorems on factorization and primality testing," in *Proc. Cambridge Philos. Soc.*, vol. 76, pp. 521–528, 1974.
[23] J. J. Quisquater and C. Couvreur, "Fast decipherment algorithm for RSA public key cryptosystem," *Electronics Letters*, vol. 18, no. 1, pp. 905–907, 1982.
[24] J. J. Quisquater, "How to avoid successful cryptanalysis of your random prime generator (abstract)," in *Presented at the conference of cryptography*, CIRM Lyuminy, Framce, Sep. 1995.
[25] R. L. Rivest and A. Shamir, "Efficient factoring based on partial information," in *CRYPTO'85*, LNCS 219, pp. 31–34, Springer-Verlag, 1986.
[26] R. L. Rivest and R. D. Silverman, *Are Strong Primes Needed for RSA*, The 1997 RSA Laboratories Seminar Series, Seminars Proceeding, 1997.
[27] R. L. Ravist, A. Shamir, and L. Adllemann, "A method for obtaining digital signature and public key cryptosystems," *Commmunications Of the ACM*, vol. 21, vol. 2, pp. 120–126, 1978.
[28] J. H. Silverman, *The Arithmetic of Elliptic Curve*, Graduate text in mathematics, vol. 106, Springer Berlin, 1986.
[29] R. D. Silverman, "Fast generation of random, strong RSA primes," *CrptoBytes*, vol. 3, no. 1, pp. 9–13, 1997.
[30] G. J. Simmons, "A weak privacy protocol using the RSA cryptoalgorithm," *Cryptologia*, vol. 7, no. 2, pp. 180–182, 1983.
[31] S. K. Chua, K. H. Leung, S. Ling, "Attack on RSA-type cryptosystem based on singular cubic curves over Z/nZ^*," *Theoretical Computer Science*, vol. 220, pp. 19–27, 1999.
[32] G. J. Simmons and M. J Norris, "Preliminary comment on MIT public key cryptosystem," *Crypologia*, vol. 1, pp. 406–414, 1977.
[33] M. J. Wiener, "Cryptanalysis of short RSA secret exponents," *IEEE Transactions on Information Theory*, vol. IT-36, pp. 386-396, 1990.
[34] H. C. Williams, "A p+1 method of factoring," *Mathematics of Computation*, vol. 39, no. 159, pp. 225–234, 1982.

Sahadeo Padhye received the B.Sc. and M.Sc. degree in Mathematics form Pt. Ravishankar Shukla University, Raipur. Chhattisgarh, India in 1999 and 2001. Council of Scientific and Industrial Research (CSIR), India has granted him Junior Research Fellowship (2002-2004). He then joined School of Studies in Mathematics, Pt. Ravishankar Shukla University, Raipur, India for his research work. He is a life member of Cryptology Research Society of India (CRSI). His area of interest is Public Key Cryptography based on elliptic curve.