SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2016, Vol. 59 032108:1–032108:10
doi: 10.1007/s11432-015-5325-7

Generalized cryptanalysis of RSA with small public exponent

Mengce ZHENG, Honggang HU* & Zilong WANG

Key Laboratory of Electromagnetic Space Information, Chinese Academy of Sciences (CAS), School of Information Science and Technology, University of Science and Technology of China, Hefei 230027*, China*

Received September 4, 2015; accepted December 27, 2015; published online January 15, 2016

Abstract In this paper, we demonstrate that there exist weak keys in the RSA public-key cryptosystem with the public exponent $e = N^{\alpha} \leqslant N^{0.5}$. In 1999, Boneh and Durfee showed that when $\alpha \approx 1$ and the private exponent $d = N^{\beta} < N^{0.292}$, the system is insecure. Moreover, their attack is still effective for $0.5 < \alpha < 1.875$. We propose a generalized cryptanalytic method to attack the RSA cryptosystem with $\alpha \leqslant 0.5$. For $c = \lfloor \frac{1-\alpha}{\alpha} \rfloor$ and $e^{\gamma c} \equiv d \pmod{e^c}$, when γ, β satisfy $\gamma < 1 + \frac{1}{c} - \frac{1}{2\alpha c}$ and $\beta < \alpha c + \frac{7}{6} - \alpha\gamma c - \frac{1}{3}\sqrt{6\alpha + 6\alpha c + 1 - 6\alpha\gamma c}$, we can perform cryptanalytic attacks based on the LLL algorithm. The basic idea is an application of Coppersmith's techniques and we further adapt the technique of unravelled linearization, which leads to an optimized lattice. Our advantage is that we achieve new attacks on RSA with $\alpha \leqslant 0.5$ and consequently, there exist weak keys in RSA for most α.

Keywords cryptanalysis, RSA, LLL algorithm, Coppersmith's techniques, unravelled linearization

Citation Zheng M C, Hu H G, Wang Z L. Generalized cryptanalysis of RSA with small public exponent. Sci China Inf Sci, 2016, 59(3): 032108, doi: 10.1007/s11432-015-5325-7

1 Introduction

The RSA cryptosystem [1] plays an important role in the area of information security due to its popularity. Many researchers have studied its vulnerability in various cases such as small public exponent [2–4], small private exponent [5–8], small private CRT-exponent [9–12], partial key exposure [13–17], etc. Since Coppersmith introduced a new method of finding small roots of modular equations [2, 3], its variations have been widely used in the field of cryptanalysis of RSA, of which the most well-known and useful one is Boneh-Durfee attack [6, 7].

In the case of RSA, the modulus $N = pq$ is the product of two primes with the same bit length. (N, e) denotes the public key and (p, q, d) denotes the private key. As we know, the main equation of RSA is $ed \equiv 1 \pmod{\phi(N)}$, where $\phi(N)$ is Euler's totient function. Since $\phi(N) = (p-1)(q-1)$, the following relation for some integers y exists,

$$ed + y((N+1) - (p+q)) = 1. \tag{1}$$

Eq. (1) can be transformed into $y(A + z) \equiv 1 \pmod{e}$ for $A = N + 1$ and $z = -(p + q)$.

* Corresponding author (email: hghu2005@ustc.edu.cn)

Throughout the paper we write $e = N^{\alpha}$ and $d = N^{\beta}$. On one hand, we have $|y| < \frac{de-1}{\phi(N)} \approx N^{\alpha+\beta-1}$ by (1). On the other hand, it is expected to be secure when $p < q < 2p$. So it implies that $p < \sqrt{N}$ and $q < 2\sqrt{N}$. Similarly, we have $|z| < 3N^{\frac{1}{2}}$. In the common case, we take $\alpha \approx 1$ and ignore the constants. Hence the problem is to find integers y and z satisfying the following modular equation

$$y(A+z) - 1 \equiv 0 \pmod{e}, \tag{2}$$

where $|y| < N^{\alpha+\beta-1}$ and $|z| < N^{\frac{1}{2}}$.

Boneh and Durfee showed that for $\beta < 1 - \frac{\sqrt{2}}{2} \approx 0.292$, the RSA cryptosystem is insecure. Moreover, for $0.5 < \alpha < 1.875$, they still achieved good results. Unfortunately, Boneh-Durfee attack does not work any more for $\alpha \leqslant 0.5$. To improve the bound of α, Luo et al. [18] studied the special case of $d > e$. They showed that it is insecure when $0.258 \leqslant \alpha \leqslant 0.854$ and some other conditions are satisfied. We briefly mention their method from which our new improvement is derived. Taking $x \equiv d \pmod{e}$ and we assume $x \approx e^{\gamma} = N^{\alpha\gamma}$. Then we obtain a new relation similar to (2) from (1) by replacing d with x, $ex + y(A+z) \equiv 1 \pmod{e^2}$. Now the problem turns to finding integers x, y, and z satisfying

$$ex + y(A+z) - 1 \equiv 0 \pmod{e^2}, \tag{3}$$

where $|x| < N^{\alpha\gamma}$, $|y| < N^{\alpha+\beta-1}$, and $|z| < N^{\frac{1}{2}}$.

However, their attack also fails for $\alpha < 0.258$. Because a small public exponent is usually used in practice, we generalize the former attack to find more weak keys in RSA for $\alpha \leqslant 0.5$. In order to achieve a much better bound of α, we firstly focus on the situation when e decreases and let $x \equiv d \pmod{e^c}$ for $c = \lfloor \frac{1-\alpha}{\alpha} \rfloor$. It means that we round down c to the integer portion of $\frac{1-\alpha}{\alpha}$. Assume that $x \approx e^{\gamma c} = N^{\alpha\gamma c}$ and hence the following trivariate modular equation that is similar to (3) exists,

$$ex + y(A+z) - 1 \equiv 0 \pmod{e^{c+1}}, \tag{4}$$

where $|x| < N^{\alpha\gamma c}$, $|y| < N^{\alpha+\beta-1}$, and $|z| < N^{\frac{1}{2}}$.

Secondly, we adapt the technique of unravelled linearization [19–21] to optimize the above modular equation. This technique can further improve the bound of β and reduce the dimension of the lattice at the same time. We glue the monomials yz and 1 together and denote $(yz-1)$ by a new variable u. Hence we obtain the optimized linear modular equation,

$$ex + Ay + u \equiv 0 \pmod{e^{c+1}}, \tag{5}$$

where $|x| < N^{\alpha\gamma c}$, $|y| < N^{\alpha+\beta-1}$, and $|u| < N^{\alpha+\beta-\frac{1}{2}}$.

The rest of the paper is organized as follows: We review preliminaries and state basic results from lattice reduction theory in Section 2. In Section 3, the method of taking $x \equiv d \pmod{e^c}$ for $c = \lfloor \frac{1-\alpha}{\alpha} \rfloor$ and $\alpha \leqslant 0.5$, which is the foundation of our work will be described. In Section 4, we use the technique of unravelled linearization to obtain a larger bound of β and give experimental results. The paper concludes in Section 5.

2 Preliminaries

A lattice $\mathcal{L}$ spanned by $b_1, \ldots, b_m$, which are linearly independent vectors in $\mathbb{R}^n$ is the set of all integer linear combinations of these vectors and $(b_1, \ldots, b_m)$ is called a basis of $\mathcal{L}$. m is called the dimension of the lattice and it is called full-rank if $m = n$. $\mathcal{L}$ can be denoted by

$$\mathcal{L}(b_1, \ldots, b_m) = \left\{ \sum_{i=1}^{m} z_i b_i | z_i \in \mathbb{Z} \right\}.$$

For $i = 1, \ldots, m$, we regard each vector b_i as a row vector and they generate the $m \times n$ matrix B. Thus the determinant of lattice $\mathcal{L}$ is defined as $\det(\mathcal{L}) = \sqrt{\det(BB^{\mathrm{T}})}$. We have $\det(\mathcal{L}) = \det(B)$ when

$\mathcal{L}$ is full-rank and it implies that B is a square matrix. It can be easily inferred that different bases of a lattice do not change its determinant. Therefore, we provide another definition of the determinant $\det(\mathcal{L}) = \prod_{i=1}^{m} \|b_i^*\|$, where $b_1^*, \ldots, b_m^*$ are derived from Gram-Schmidt orthogonalization to the basis $(b_1, \ldots, b_m)$, and $\|\cdot\|$ denotes the Euclidean norm of a vector.

The LLL algorithm proposed by Lenstra, Lenstra, and Lovász [22] is widely used for many applications due to its efficient results. We provide the following substratal lemma about the running results of the LLL algorithm from their paper.

Lemma 1 (LLL)**.** Let $\mathcal{L}$ be a lattice spanned by a basis $(b_1, b_2, \ldots, b_m)$. The LLL algorithm gives an output of a reduced basis $(v_1, v_2, \ldots, v_m)$ of $\mathcal{L}$, that satisfies

1. $\|v_i\|^2 \leqslant 2^{j-1}\|v_j^*\|^2,\ 1 \leqslant i \leqslant j \leqslant m$,
2. $\|v_1\| \leqslant 2^{\frac{m-1}{4}} \det(\mathcal{L})^{\frac{1}{m}}$.

We also can compute the bounds of other vectors in the LLL-reduced basis except v_1 in the Euclidean norm. We show the following lemma for a basis $(b_1, b_2, \cdots, b_m)$ of $\mathcal{L}$ and an auxiliary parameter $b_{\min}^* = \min_i \|b_i^*\|$.

Lemma 2. Let $\mathcal{L}$ be a lattice spanned by a basis $(b_1, b_2, \ldots, b_m)$ and suppose that $b_{\min}^* \geqslant 1$. The LLL algorithm outputs a reduced basis $(v_1, v_2, \ldots, v_m)$ of $\mathcal{L}$, and v_i satisfies

$$\|v_i\| \leqslant 2^{\frac{m+i-2}{4}} \det(\mathcal{L})^{\frac{1}{m+1-i}}.$$

Proof. We have $\|v_i^*\| \geqslant b_{\min}^* \geqslant 1$. By the first part of Lemma 1, for $2 \leqslant i \leqslant m$ we have

$$\|v_i\|^{2(m+1-i)} \leqslant \prod_{j=i}^{m} 2^{j-1}\|v_j^*\|^2 = 2^{\frac{(m+i-2)(m+1-i)}{2}} \cdot \frac{\det(\mathcal{L})^2}{\prod_{j=1}^{i-1}\|v_j^*\|^2} \leqslant 2^{\frac{(m+i-2)(m+1-i)}{2}} \det(\mathcal{L})^2.$$

Thus combining it with the second part of Lemma 1 for $i = 1$ directly leads to Lemma 2.

We only need $i = 1, 2, 3$ in Lemma 2, namely $\|v_1\|, \|v_2\|, \|v_3\| \leqslant 2^{\frac{m+1}{4}} \det(\mathcal{L})^{\frac{1}{m-2}}$, and it is sufficient for our application. The following lemma presented by Howgrave-Graham [4] gives a judging condition when the roots of a modular equation in a sufficiently small norm are also roots over the integers. Then we can combine Lemma 2 with Lemma 3 to solve a modular equation. To a given polynomial $g(x, y, z) = \sum_{i,j,k} a_{i,j,k} x^i y^j z^k$, its norm is defined by $\|g(x, y, z)\|^2 = \sum_{i,j,k} |a_{i,j,k}|^2$.

Lemma 3 (Howgrave-Graham)**.** Let $g(x, y, z) = \sum_{i,j,k} a_{i,j,k} x^i y^j z^k \in \mathbb{Z}[x, y, z]$ be a polynomial that is a sum of most m monomials and $g(xX, yY, zZ) = \sum_{i,j,k} a_{i,j,k} X^i Y^j Z^k x^i y^j z^k$ for given X, Y, and Z. Suppose that

1. $g(x_0, y_0, z_0) \equiv 0 \pmod{R}$, where $|x_0| < X, |y_0| < Y$, and $|z_0| < Z$,
2. $\|g(xX, yY, zZ)\| < \frac{R}{\sqrt{m}}$.

Then the equation also holds over the integers, namely $g(x_0, y_0, z_0) = 0$.

Proof. By applying Cauchy inequality, we know that

$$\begin{aligned} |g(x_0, y_0, z_0)| &= \left|\sum_{i,j,k} a_{i,j,k} x_0^i y_0^j z_0^k\right| < \left|\sum_{i,j,k} a_{i,j,k} X^i Y^j Z^k\right| \leqslant \sum_{i,j,k} 1 \cdot \left|a_{i,j,k} X^i Y^j Z^k\right| \\ &\leqslant \sqrt{m} \cdot \|g(xX, yY, zZ)\| < R, \end{aligned}$$

and combine it with the first part, by which we obtain $g(x_0, y_0, z_0) = 0$.

3 Solving trivariate modular equations

According to Coppersmith's techniques, the basic idea for finding small roots of modular equations is to reduce this problem to finding roots over the integers. To do so, we construct a set of polynomials with a common root modulo e^{c+1}. The extreme case that $d < N^{1-\alpha}$ occurs with negligible probability, so we have $d > e^c$ for taking $c = \lfloor \frac{1-\alpha}{\alpha} \rfloor$. Then we begin to search an integer linear combination of

the constructed polynomials' coefficient vectors by the LLL algorithm and its norm is expected to be sufficiently small.

To solve (4), We define $f(x, y, z) = ex + y(A + z) - 1$, and it turns to finding integers (x_0, y_0, z_0) satisfying the following trivariate modular equation,

$$f(x_0, y_0, z_0) \equiv 0 \pmod{e^{c+1}},$$

where $|x_0| < N^{\alpha\gamma c}$, $|y_0| < N^{\alpha+\beta-1}$, and $|z_0| < N^{\frac{1}{2}}$.

We start with the above trivariate modular equation. As described previously, we need to define suitable polynomials that share the common root (x_0, y_0, z_0) in order to obtain an integer linear combination. Therefore, we define the polynomials for some positive integers s and t that will be determined later,

$$\begin{aligned} g_{i,j,k}(x, y, z) &= x^i y^j f^k e^{(c+1)(s-k)} \quad \text{for} \quad i + j + k = r = 0, \ldots, s, \\ h_{i,j,k}(x, y, z) &= x^i z^j f^k e^{(c+1)(s-k)} \quad \text{for} \quad j = 1, \ldots, t, \text{ and } i + k = r = 0, \ldots, s. \end{aligned}$$

The above symbol r is the sum of the exponents and we apply r to simply distinguish the blocks in the constructed matrix. We further explain the meaning of $g_{i,j,k}$ and $h_{i,j,k}$. $i + j + k = r = 0, \ldots, s$ implies that $i = 0, \ldots, s, j = 0, \ldots, s - i$, and $k = 0, \ldots, s - i - j$. Similarly, $i + k = r = 0, \ldots, s$ implies that $i = 0, \ldots, s$, and $k = 0, \ldots, s - i$.

It is obvious that (x_0, y_0, z_0) is the common root of all the polynomials $g_{i,j,k}$ and $h_{i,j,k}$. Afterwards, we begin to search an integer linear combination of all $g_{i,j,k}(xX, yY, zZ)$ and $h_{i,j,k}(xX, yY, zZ)$ by the LLL algorithm and ensure that its norm is sufficiently small in order to meet the conditions in Lemma 3. Here we know that $X = N^{\alpha\gamma c}$, $Y = N^{\alpha+\beta-1}$, and $Z = N^{\frac{1}{2}}$. However, there is a slight difference that we take $e^{(c+1)s}$ as the modulus. Then we build a lattice $\mathcal{L}$ spanned by the corresponding coefficient vectors and use the LLL algorithm to find a small norm vector that will yield a small norm polynomial. Lattice $\mathcal{L}$ can be represented by a square matrix whose rows are the polynomials' coefficient vectors.

We divide the matrix into two big blocks G, H and the corresponding coefficient vectors derived from $g_{i,j,k}$ and $h_{i,j,k}$ are contained. The following three restrictions are provided when we construct the matrix that generates $\mathcal{L}$:

1. G blocks are set higher than H blocks. Each monomial that appears in $g_{i,j,k}$ and $h_{i,j,k}$ is considered to be a component of the row vector and arrayed in an appropriate order. As each row vector introduces a new monomial according to the following two restrictions, the order of these monomials is the order of the occurrence of each new monomial.

2. In G blocks, each row vector is denoted by the triple (i, j, k). Moreover, we divide G into several small blocks (between two horizontal lines in Table 1) according to increasing r's. In each small block with a fixed r, we arrange (i, j, k) with larger i's and smaller k's to appear in higher rows.

3. In H blocks, each row vector is also denoted by the triple (i, j, k). Moreover, we divide H into several small blocks (also between two horizontal lines in Table 1) according to increasing t's and r's. We notice that r is different from that in G blocks. In each small block with fixed t and r, we arrange (i, j, k) with smaller j's and smaller k's to appear in higher rows.

A simple example is shown in Table 1 and other non-zero entries are denoted by $*$.

To the parameters s and t, we assume $t = \tau s$ for easy calculation. Then we can compute the dimension of the full-rank lattice, which is denoted by m. After that, our task is to compute the determinant of $\mathcal{L}$. Since it is a lower triangular square matrix, we can easily compute the determinant by counting the numbers of X, Y, Z, and E in the diagonal entries, respectively.

$$\begin{aligned} m &= \sum_{r=0}^{s}\sum_{i=0}^{r}\sum_{j=0}^{r-i} 1 + \sum_{j=1}^{t}\sum_{r=0}^{s}\sum_{i=0}^{r} 1 = \binom{s+3}{3} + t \cdot \binom{s+2}{2} = \frac{1+3\tau}{6}s^3 + o(s^3), \\ X_n &= \sum_{r=0}^{s}\sum_{i=0}^{r} i \sum_{j=0}^{r-i} 1 + \sum_{j=1}^{t}\sum_{r=0}^{s}\sum_{i=0}^{r} i = \binom{s+3}{4} + t \cdot \binom{s+2}{3} = \frac{1+4\tau}{24}s^4 + o(s^4), \end{aligned}$$

Table 1 A simple example with $s = 2$, $t = 1$ for $E = e^{c+1}$

	1	x	y	yz	x^2	xy	y^2	xyz	y^2z	y^2z^2	z	xz	yz^2	x^2z	xyz^2	y^2z^3
$G(0,0,0)$	E^2															
$G(1,0,0)$		E^2X														
$G(0,1,0)$			E^2Y													
$G(0,0,1)$	*	*	*	EYZ												
$G(2,0,0)$					E^2X^2											
$G(1,1,0)$						E^2XY										
$G(0,2,0)$							E^2Y^2									
$G(1,0,1)$		*			*	*		$EXYZ$								
$G(0,1,1)$			*			*	*		EY^2Z							
$G(0,0,2)$	*	*	*	*	*	*	*	*	*	Y^2Z^2						
$H(0,1,0)$											E^2Z					
$H(1,1,0)$												E^2XZ				
$H(0,1,1)$				*							*	*	EYZ^2			
$H(2,1,0)$														E^2X^2Z		
$H(1,1,1)$								*				*		*	$EXYZ^2$	
$H(0,1,2)$				*				*	*	*	*	*	*	*	*	Y^2Z^3

$$Y_n = \sum_{r=0}^{s}\sum_{j=0}^{r} j \sum_{i=0}^{r-j} 1 + \sum_{r=0}^{s}\sum_{k=0}^{r} k \sum_{i=0}^{r-k} 1 + \sum_{j=1}^{t}\sum_{r=0}^{s}\sum_{k=0}^{r} k = 2 \cdot \binom{s+3}{4} + t \cdot \binom{s+2}{3} = \frac{1+2\tau}{12} s^4 + o(s^4),$$

$$Z_n = \sum_{r=0}^{s}\sum_{k=0}^{r} k \sum_{i=0}^{r-k} 1 + \sum_{j=1}^{t} j \sum_{r=0}^{s}\sum_{i=0}^{r} i + \sum_{j=1}^{t}\sum_{r=0}^{s}\sum_{k=0}^{r} k = \binom{s+3}{4} + \binom{t+1}{2} \cdot \binom{s+2}{2} + t \cdot \binom{s+2}{3}$$

$$= \frac{1+6\tau^2+4\tau}{24} s^4 + o(s^4),$$

$$E_n = \sum_{r=0}^{s}\sum_{k=0}^{r} (s-k) \sum_{i=0}^{r-k} 1 + \sum_{j=1}^{t}\sum_{r=0}^{s}\sum_{k=0}^{r} (s-k) = 3 \cdot \binom{s+3}{4} + 2t \cdot \binom{s+2}{3} = \frac{3+8\tau}{24} s^4 + o(s^4).$$

Consequently we have $\det(\mathcal{L}) = X^{X_n} Y^{Y_n} Z^{Z_n} E^{E_n}$ for $X = N^{\alpha\gamma c}$, $Y = N^{\alpha+\beta-1}$, $Z = N^{\frac{1}{2}}$, and $E = e^{c+1}$. If $\det(\mathcal{L}) < \frac{e^{(m-2)(c+1)s}}{\eta}$ for $\eta = 2^{\frac{(m-2)(m+1)}{4}} m^{\frac{m-2}{2}}$, then the norms of v_1, v_2, and v_3 that are the first three vectors of the output LLL-reduced basis are less than $\frac{e^{(c+1)s}}{\sqrt{m}}$. Thus we can apply Lemmas 2 and 3. Once the conditions in Lemma 3 are satisfied, the corresponding polynomials g_1, g_2, and $g_3 \in \mathbb{Z}(x, y, z)$ can be obtained and three equations $g_1(x, y, z) = 0$, $g_2(x, y, z) = 0$, and $g_3(x, y, z) = 0$ hold. Then we compute their resultants $g_4(y, z) = \mathrm{Res}(g_1, g_2)$, $g_5(y, z) = \mathrm{Res}(g_1, g_3)$, and $g_6(z) = \mathrm{Res}(g_4, g_5)$.

Our method relies on the following heuristic assumption for computations with multivariate equations similar to Boneh-Durfee attack.

Assumption 1. g_1, g_2, and g_3 outputed by our lattice-based method are algebraically independent.

Although the polynomials g_1, g_2, and g_3 derived from the LLL algorithm are linearly independent, they may have a common factor. Assumption 1 ensures that we can compute the common root. Additionally, we can easily compute it by known numerical methods. Furthermore, it implies that the run time of computing the common root is negligible compared to the whole lattice-based construction. Thus if Assumption 1 is confirmed, we can solve $g_6(z) = 0$ and obtain $-(p + q)$, which leads to the factorization of N.

Now we estimate β and γ for a given α and corresponding c. Since η is negligible compared to $e^{(m-2)(c+1)s}$ when taking $s \to \infty$, its effect can be ignored. Thus it indicates that the following inequality

holds from above,

$$\alpha\gamma cX_n + (\alpha+\beta-1)Y_n + \frac{1}{2}Z_n + \alpha(c+1)E_n < \alpha(m-2)(c+1)s.$$

By taking $s \to \infty$, it can be simplified to

$$6\tau^2 + \tau(8\alpha\gamma c + 8\beta - 8\alpha c - 4) + 2\alpha\gamma c + 2\alpha + 4\beta - 2\alpha c - 3 < 0. \tag{6}$$

The value of the left part of (6) reaches its minimum when we take $\tau = \frac{2\alpha c+1-2\alpha\gamma c-2\beta}{3}$. And then we put it in to obtain the minimum value of the left part,

$$-\frac{8}{3}\beta^2 + \frac{4}{3}(4\alpha c + 5 - 4\alpha\gamma c)\beta + 2\alpha\gamma c + 2\alpha - 2\alpha c - 3 - \frac{2}{3}(2\alpha\gamma c - 2\alpha c - 1)^2 < 0.$$

Then we have

$$\beta < \alpha c + \frac{5}{4} - \alpha\gamma c - \frac{\sqrt{3}}{4}\sqrt{4\alpha + 4\alpha c + 1 - 4\alpha\gamma c}. \tag{7}$$

For the case when we take $c = 1$, this solution is same as that in [18].

As we compute the bounds of β and γ, the following additional conditions are considered: $1-\alpha < \beta$, $0 \leqslant 2\alpha c + 1 - 2\alpha\gamma c - 2\beta$, and $0 \leqslant 4\alpha + 4\alpha c + 1 - 4\alpha\gamma c$. Thus we have $1-\alpha < \beta \leqslant \alpha c - \alpha\gamma c + \frac{1}{2}$. Combine it with (7) and then we obtain the bounds of β and γ,

$$\begin{aligned}
&\beta < \alpha c + \frac{5}{4} - \alpha\gamma c - \frac{\sqrt{3}}{4}\sqrt{4\alpha + 4\alpha c + 1 - 4\alpha\gamma c},\\
&\gamma < 1 + \frac{1}{c} - \frac{1}{2\alpha c}.
\end{aligned}$$

We show the bounds of β and γ with particular α compared with the results obtained by the optimized linear modular equation in next section. Therefore, we observe that RSA with $e \leqslant N^{0.5}$ is vulnerable to lattice-based attacks for ideal conditions.

4 Solving optimized linear modular equations

In this section, we focus on (5) that is optimized by unravelled linearization. The effect of the new method is to capture the sublattice structure of the lattice constructed in the fundamental attack in Section 3. Hence similar to Section 3, we also define the following polynomial $\bar{f}(x, y, u) = ex + Ay + u$, and it turns to finding integers (x_0, y_0, u_0) satisfying the following linear modular equation,

$$\bar{f}(x_0, y_0, u_0) \equiv 0 \pmod{e^{c+1}},$$

where $|x_0| < N^{\alpha\gamma c}$, $|y_0| < N^{\alpha+\beta-1}$, and $|u_0| < N^{\alpha+\beta-\frac{1}{2}}$.

Now we define suitable polynomials that share a common root in order to obtain an integer linear combination. To do so, we define the polynomials for positive integers s and t ($t \leqslant s$), which will be determined later,

$$\begin{aligned}
&\bar{g}_{i,j,k}(x, y, u) = x^i y^j \bar{f}^k e^{(c+1)(s-k)} \quad \text{for} \quad i+j+k = r = 0, \ldots, s,\\
&\bar{h}_{i,j,k}(x, z, u) = x^i z^j \bar{f}^k e^{(c+1)(s-k)} \quad \text{for} \quad j = 1, \ldots, t, \text{ and } i+k = r = \left\lfloor \frac{s}{t} \right\rfloor j, \ldots, s.
\end{aligned}$$

Since $u_0 = y_0 z_0 - 1$, (x_0, y_0, u_0) is equivalent to (x_0, z_0, u_0). They are the common roots that we need to solve. We then apply the LLL algorithm to search an integer linear combination of $\bar{g}_{i,j,k}(xX, yY, uU)$ and $\bar{h}_{i,j,k}(xX, zZ, uU)$. To do so, we build a lattice spanned by the corresponding coefficient vectors. The lattice can be represented by a matrix whose rows are the corresponding polynomials' coefficient vectors. As f turns to $\bar{f}$, we denote $U = e^{\alpha+\beta-\frac{1}{2}}$ while X, Y, Z, and E stay the same. The above example shown in Table 1 turns to be the following matrix in Table 2 and other non-zero entries are denoted by $*$ as well.

Table 2 The same example with $s = 2$, $t = 1$ for $E = e^{c+1}$ by taking $u = yz - 1$

	1	x	y	u	x^2	xy	y^2	xu	yu	u^2	x^2z	xzu	zu^2
$\bar{G}(0,0,0)$	E^2												
$\bar{G}(1,0,0)$		E^2X											
$\bar{G}(0,1,0)$			E^2Y										
$\bar{G}(0,0,1)$		*	*	EU									
$\bar{G}(2,0,0)$					E^2X^2								
$\bar{G}(1,1,0)$						E^2XY							
$\bar{G}(0,2,0)$							E^2Y^2						
$\bar{G}(1,0,1)$					*	*		EXU					
$\bar{G}(0,1,1)$						*	*		EYU				
$\bar{G}(0,0,2)$					*	*	*	*	*	U^2			
$\bar{H}(2,1,0)$											E^2X^2Z		
$\bar{H}(1,1,1)$		*						*			*	$EXZU$	
$\bar{H}(0,1,2)$		*	*	*				*	*	*	*	*	ZU^2

The method to divide the matrix into two blocks $\bar{G}$ and $\bar{H}$ is alike, so we omit it. We observe that the optimized matrix is still a lower triangular square matrix and hence it generates a full-rank lattice. A detailed analysis will be given in the appendix. To the parameters s and t, we assume $t = \tau s$ and $\lfloor \frac{s}{t} \rfloor = \frac{1}{\tau}$ for simplicity. Hence, we compute the dimension $\bar{m}$ and the numbers of exponents of X, Y, Z, U, and E elements on the diagonal, respectively.

$$\bar{m} = \sum_{r=0}^{s}\sum_{i=0}^{r}\sum_{j=0}^{r-i} 1 + \sum_{j=1}^{t}\sum_{r=j/\tau}^{s}\sum_{i=0}^{r} 1 = \frac{1+2\tau}{6}s^3 + o(s^3),$$

$$\bar{X}_n = \sum_{r=0}^{s}\sum_{i=0}^{r} i\sum_{j=0}^{r-i} 1 + \sum_{j=1}^{t}\sum_{r=j/\tau}^{s}\sum_{i=0}^{r} i = \frac{1+3\tau}{24}s^4 + o(s^4),$$

$$\bar{Y}_n = \sum_{r=0}^{s}\sum_{j=0}^{r} j\sum_{i=0}^{r-j} 1 = \frac{1}{24}s^4 + o(s^4),$$

$$\bar{Z}_n = \sum_{j=1}^{t} j\sum_{r=j/\tau}^{s}\sum_{i=0}^{r} i = \frac{\tau^2}{8}s^4 + o(s^4),$$

$$\bar{U}_n = \sum_{r=0}^{s}\sum_{k=0}^{r} k\sum_{i=0}^{r-k} 1 + \sum_{j=1}^{t}\sum_{r=j/\tau}^{s}\sum_{k=0}^{r} k = \frac{1+3\tau}{24}s^4 + o(s^4),$$

$$\bar{E}_n = \sum_{r=0}^{s}\sum_{k=0}^{r}(s-k)\sum_{i=0}^{r-k} 1 + \sum_{j=1}^{t}\sum_{r=j/\tau}^{s}\sum_{k=0}^{r}(s-k) = \frac{3+5\tau}{24}s^4 + o(s^4).$$

Now we estimate β and γ for a given α and corresponding c. Follow the analysis in Section 3 and thus the following inequality holds,

$$\alpha\gamma c\bar{X}_n + (\alpha+\beta-1)\bar{Y}_n + \frac{1}{2}\bar{Z}_n + (\alpha+\beta-\frac{1}{2})\bar{U}_n + \alpha(c+1)\bar{E}_n < \alpha(m-2)(c+1)s.$$

By taking $s \to \infty$, it can be simplified to

$$3\tau^2 + \tau(6\alpha\gamma c + 6\beta - 6\alpha c - 3) + 2\alpha\gamma c + 2\alpha + 4\beta - 2\alpha c - 3 < 0. \tag{8}$$

The value of the left part of (8) reaches its minimum when we take $\tau = \frac{2\alpha c+1-2\alpha\gamma c-2\beta}{2}$. Then we plug it in to obtain the minimum value of the left part,

$$-3\beta^2 + (6\alpha c + 7 - 6\alpha\gamma c)\beta + 2\alpha\gamma c + 2\alpha - 2\alpha c - 3 - \frac{3}{4}(2\alpha\gamma c - 2\alpha c - 1)^2 < 0.$$

Table 3 Comparison with previous results on the theoretical bounds of β and γ with particular α

α	c	$\gamma_{\max}$	$\beta_{\min}$	$\beta_{\max}^{\gamma=0}$ of Sect. 3	$\beta_{\max}^{\gamma=0}$ of Sect. 4	$\beta_{\max}^{\gamma=0}$ of [18]
0.01	99	0.505	0.99	1.272	1.275	—
0.05	19	0.526	0.95	1.232	1.235	—
0.15	5	0.533	0.85	1.071	1.073	—
0.25	3	0.666	0.75	1.032	1.035	0.75
0.3	2	0.666	0.7	0.921	0.923	0.752
0.4	1	0.75	0.6	0.763	0.764	0.763
0.5	1	1	0.5	0.782	0.785	0.782

Then we have

$$\beta < \alpha c + \frac{7}{6} - \alpha\gamma c - \frac{1}{3}\sqrt{6\alpha + 6\alpha c + 1 - 6\alpha\gamma c}. \tag{9}$$

Similarly, we consider $1-\alpha < \beta$, $0 \leqslant 2\alpha c + 1 - 2\alpha\gamma c - 2\beta \leqslant 2$, and $0 \leqslant 6\alpha + 6\alpha c + 1 - 6\alpha\gamma c$. Thus we know that $1 - \alpha < \beta \leqslant \alpha c - \alpha\gamma c + \frac{1}{2}$. Combine it with (9) and then we obtain the bounds of β and γ,

$$\beta < \alpha c + \frac{7}{6} - \alpha\gamma c - \frac{1}{3}\sqrt{6\alpha + 6\alpha c + 1 - 6\alpha\gamma c},$$
$$\gamma < 1 + \frac{1}{c} - \frac{1}{2\alpha c}.$$

Table 3 shows the bounds of β and γ with particular α. The comparison between our results and previous results in [18] is also showed in Table 3. Thus we observe that the bound of β derived from our new method with unravelled linearization is larger than that in Section 3. Moreover, we point out that this new method can reduce the dimension of the lattice from $m = \frac{s^3+3ts^2}{6}$ to $\bar{m} = \frac{s^3+2ts^2}{6}$. As s and t become larger, it can significantly simplify the amount of calculation in the LLL algorithm.

We did several experiments to check if the assumption holds. These experiments were performed under Ubuntu 14.04 running on a computer with Intel(R) Core(TM) i5-3210M CPU 2.50 GHz, 3 GB RAM and 3 MB Cache. We carried out the experiments by using the LLL implementation available in the NTL library. The numbers used in each experiment were chosen uniformly at random and the previously mentioned conditions were satisfied.

When we choose $s = 2$ and $t = 1$, the dimension of the lattice is 13. It takes 2.852 s to run the LLL implementation for a 1024 bit modulus.

When we choose $s = 3$ and $t = 1$, the dimension of the lattice is 24. It takes 1108.37 s to run the LLL implementation for a 1024 bit modulus. We obtain three independent polynomials $g_1(x, y, z)$, $g_2(x, y, z)$, and $g_3(x, y, z)$ from the output. Then we can solve the common root by computing the resultants and they are the desired correct results. Although we cannot prove that our attack always succeeds, our experiments confirm Assumption 1.

5 Conclusion

We show that in some circumstances we can perform lattice-based attacks on the RSA cryptosystem with the public exponent $e \leqslant N^{0.5}$. We note that $e = N^{\alpha}$, $d = N^{\beta}$, and $e^{\gamma c} \equiv d \pmod{e^c}$ for $c = \lfloor \frac{1-\alpha}{\alpha} \rfloor$, if $\gamma < 1 + \frac{1}{c} - \frac{1}{2\alpha c}$ and $\beta < \alpha c + \frac{7}{6} - \alpha\gamma c - \frac{1}{3}\sqrt{6\alpha + 6\alpha c + 1 - 6\alpha\gamma c}$ hold, RSA may be insecure.

Our work is an application of Coppersmith's techniques and also an extension of Boneh-Durfee attack. We apply it to trivariate modular polynomials and further improve the bounds by unravelled linearization, which can be used to disclose the hidden sublattice structure of the original lattice. However the results of our method are still heuristic unless Assumption 1 is confirmed. So a way to handle this heuristic assumption is still an open problem.

Acknowledgements This work was supported partially by National Natural Science Foundation of China (Grant No. 61271271), 100 Talents Program of Chinese Academy of Sciences, and Fundamental Research Funds for the Central Universities in China (Grant No. WK2101020005).

Conflict of interest The authors declare that they have no conflict of interest.

References

1 Rivest R L, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Commun ACM, 1978, 21: 120–126
2 Coppersmith D. Finding a small root of a univariate modular equation. In: Proceedings of International Conference on the Theory and Application of Cryptographic Techniques, Saragossa, 1996. 155–165
3 Coppersmith D. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. J Cryptol, 1997, 10: 233–260
4 Howgrave-Graham N. Finding small roots of univariate modular equations revisited. In: Darnell M, ed. Crytography and Coding. Berlin: Springer, 1997. 131–142
5 Wiener M J. Cryptanalysis of short RSA secret exponents. IEEE Trans Inform Theory, 1990, 36: 553–558
6 Boneh D, Durfee G. Cryptanalysis of RSA with private key d less than $N^{0.292}$. In: Proceedings of International Conference on the Theory and Application of Cryptographic Techniques, Prague, 1999. 1–11
7 Boneh D, Durfee G. Cryptanalysis of RSA with private key d less than $N^{0.292}$. IEEE Trans Inform Theory, 2000, 46: 1339–1349
8 Blömer J, May A. Low secret exponent RSA revisited. In: Silverman J H, ed. Cryptography and Lattices. Berlin: Springer, 2001. 4–19
9 May A. Cryptanalysis of unbalanced RSA with small CRT-exponent. In: Proceedings of 22nd Annual International Cryptology Conference, Santa Barbara, 2002. 242–256
10 Jochemsz E, May A. A strategy for finding roots of multivariate polynomials with new applications in attacking RSA variants. In: Proceedings of 12th International Conference on the Theory and Application of Cryptology and Information Security, Shanghai, 2006. 267–282
11 Bleichenbacher D, May A. New attacks on RSA with small secret CRT-exponents. In: Proceedings of 9th International Conference on Theory and Practice of Public-Key Cryptography, New York, 2006. 1–13
12 Jochemsz E, May A. A polynomial time attack on RSA with private CRT-exponents smaller than $N^{0.073}$. In: Proceedings of 27th Annual International Cryptology Conference, Santa Barbara, 2007. 395–411
13 Blömer J, May A. New partial key exposure attacks on RSA. In: Proceedings of 23rd Annual International Cryptology Conference, Santa Barbara, 2003. 27–43
14 Ernst M, Jochemsz E, May A, et al. Partial key exposure attacks on RSA up to full size exponents. In: Proceedings of 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, 2005. 371–386
15 Aono Y. A new lattice construction for partial key exposure attack for RSA. In: Proceedings of 12th International Conference on Practice and Theory in Public Key Cryptography, Irvine, 2009. 34–53
16 Sarkar S. Partial key exposure: generalized framework to attack RSA. In: Proceedings of 12th International Conference on Cryptology in India, Chennai, 2011. 76–92
17 Joye M, Lepoint T. Partial key exposure on RSA with private exponents larger than N. In: Ryan M D, Smyth B, Wang G L, eds. Information Security Practice and Experience. Berlin: Springer, 2012. 369–380
18 Luo P, Zhou H J, Wang D S, et al. Cryptanalysis of RSA for a special case with $d > e$. Sci China Ser-F: Inf Sci, 2009, 52: 609–616
19 Herrmann M, May A. Attacking power generators using unravelled linearization: When do we output too much? In: Proceedings of 15th International Conference on the Theory and Application of Cryptology and Information Security, Tokyo, 2009. 487–504
20 Herrmann M, May A. Maximizing small root bounds by linearization and applications to small secret exponent RSA. In: Proceedings of 13th International Conference on Practice and Theory in Public Key Cryptography, Paris, 2010. 53–69
21 Herrmann M. Lattice-based cryptanalysis using unravelled linearization. Dissertation for Doctoral Degree. Germany: Ruhr-Universität Bochum, 2011
22 Lenstra A K, Lenstra H W, Lovász L. Factoring polynomials with rational coefficients. Math Ann, 1982, 261: 515–534

Appendix A Detailed analysis of $t \leqslant s$

In order to prove that the basis matrix still keeps the triangular and square structure, we analyse its row vectors (mainly $\bar{H}(i,j,k)$). Since $u = yz - 1$, we can replace each yz by $u + 1$. It is easy to conclude that each $\bar{G}(i,j,k)$ introduces a new monomial $x^i y^j u^k$ for particular i,j,k and each $\bar{H}(i,j,0)$ introduces a new monomial $x^i z^j$ for particular i,j. Then we focus on whether or not each $\bar{H}(i,j,k)$ introduces a new monomial $x^i z^j u^k$ as well.

For the sake of convenience, we omit the multiplicative factors because they do not influence the set of monomials (components of row vectors in the matrix). Let us first think about an arbitrary $\bar{H}(i',j',k')$. We know that four families of monomials appear before $\bar{H}(i',j',k')$.

1. $x^i y^j u^k$ for $i + j + k = 0, \ldots, s$ in $\bar{G}$ blocks,
2. $x^i z^j u^k$ for $1 \leqslant j \leqslant j' - 1, i + k = \lfloor \frac{s}{t} \rfloor j, \ldots, s$ in $\bar{H}$ blocks,
3. $x^i z^j u^k$ for $j = j', i + k = \lfloor \frac{s}{t} \rfloor j', \ldots, i' + k' - 1$ in $\bar{H}$ blocks,

4. $x^i z^j u^k$ for $j = j', i + k = i' + k', k < k'$ in $\bar{H}$ blocks.

Since $\bar{f}(x, y, u) = ex + Ay + u$, we can expand $x^{i'} z^{j'} \bar{f}^{k'}$. Ignoring its multiplicative factors and binomial coefficients, $\bar{H}(i', j', k')$ consists of monomials such as $x^{i'+k_1} y^{k_2} z^{j'} u^{k_3}$ for $k_1 + k_2 + k_3 = k'$. We offer the conclusion that $\bar{H}(i', j', k')$ introduces a new monomial $x^{i'} z^{j'} u^{k'}$ $(k_1 = k_2 = 0, k_3 = k')$ and meanwhile others $(k_3 < k')$ already appear in the matrix. We discuss it in the following three cases.

1. When $1 \leqslant j' \leqslant k_2$, we rewrite that $x^{i'+k_1} y^{k_2} z^{j'} u^{k_3} = x^{i'+k_1} (u+1)^{j'} y^{k_2 - j'} u^{k_3}$ and this type of monomials that only consist of three variables x, y, and u appear in $\bar{G}$ blocks. Since the maximal sum of the exponents of x, y, and u is $i' + k_1 + j' + k_2 - j' + k_3 = i' + k' \leqslant s$, it implies that these monomials already appear in $\bar{G}$ blocks.

2. When $1 \leqslant k_2 < j'$, we have $x^{i'+k_1} y^{k_2} z^{j'} u^{k_3} = x^{i'+k_1} (u+1)^{k_2} z^{j'-k_2} u^{k_3}$ instead. Since the exponent of z is $(j' - k_2)$ that is certainly less than j', the maximal sum of the exponents of x and u is $i' + k_1 + k_2 + k_3 = i' + k' \leqslant s$, we only require that the minimal sum of the exponents of x and u satisfies $i' + k' - k_2 \geqslant \lfloor \frac{s}{t} \rfloor (j' - k_2)$. Therefore, we take the minimum value of the left side $(k_2 = j' - 1)$, and then obtain $\lfloor \frac{s}{t} \rfloor j' - (j' - 1) \geqslant \lfloor \frac{s}{t} \rfloor (j' - (j' - 1))$. Our construction is doable if $\lfloor \frac{s}{t} \rfloor \geqslant 1$ holds. We thus obtain $s \geqslant t$.

3. When $k_2 = 0$, we have $x^{i'+k_1} y^{k_2} z^{j'} u^{k_3} = x^{i'+k_1} z^{j'} u^{k_3}$ instead. Since the exponent of z is exactly j' and the sum of the exponents of x and u is $i' + k_1 + k_3 = i' + k'$, we look at k_3. We know that $k_3 < k'$, so these monomials appear in $\bar{H}$ blocks.

To summarize, the key requirement of the triangular and square structure is $t \leqslant s$. We have noted this condition and take it into consideration in our method.