New Results on Solving Linear Equations Modulo Unknown Divisors and its Applications

Yao Lu[1,2], Rui Zhang[1], and Dongdai Lin[1]

[1] State Key Laboratory of Information Security (SKLOIS)
Institute of Information Engineering (IIE)
Chinese Academy of Sciences (CAS)
[2] University of Chinese Academy of Sciences (UCAS)
lywhhit@gmail.com, {r-zhang,ddlin}@iie.ac.cn

Abstract. We revisit the problem of finding small solutions to a collection of linear equations modulo an unknown divisor p for a known composite integer N. In Asiacrypt'08, Herrmann and May introduced a heuristic algorithm for this problem, and their algorithm has many interesting applications, such as factoring with known bits problem, fault attacks on RSA signatures, etc. In this paper, we consider two variants of Herrmann-May's equations, and propose some new techniques to solve them. Applying our algorithms, we obtain a few by far the best analytical/experimental results for RSA and its variants. Specifically,

- We improve May's results (PKC'04) on small secret exponent attack on RSA variant with moduli $N = p^r q$ ($r \geq 2$).
- We extend Nitaj's result (Africacrypt'12) on weak encryption exponents of RSA and CRT-RSA.

Keywords: lattices, RSA, Coppersmith's method

1 Introduction

Lattice-based cryptanalysis is a very useful tool in various cryptographic systems, e.g., historically, it was used to break the Merkle-Hellman knapsack cryptosystem [18]. The basic idea of the lattice-based approach is that if the system parameters of the target problem can be transformed into a basis of a certain lattice, one can find some short vectors in the desired lattice using dedicated algorithms, like the LLL-algorithm [10]. One may then hope that the secret key can be recovered once the solutions from these short vectors are extracted. Although in most cases this assumption is not rigorous in theory, it usually works well in practice.

In the above approach, a key step is to construct the desired lattice. In 1997, Coppersmith [3] presented a subtle lattice construction method, and used it to find small roots of modular equations of special forms. Since then, this approach has been widely applied in the analysis of RSA. One of the most important applications is to solve approximate integer common divisor problem (ACDP), namely, given two integers that are near-multiples of a hidden integer, output that hidden integer. We note that ACDP was first introduced by Howgrave-Graham [8], which in turn has many important applications such as building fully homomorphic cryptosystems [20].

Let us briefly explain Howgrave-Graham's method. First, one reduces ACDP to solving a univariate modular polynomial:

$$f(x) = x + a \mod p$$

where a is a given integer, and p ($p \geq N^\beta$ for some $0 < \beta \leq 1$) is unknown that divides the known modulus N. Next he proposed a polynomial-time algorithm to find small roots of the univariate polynomial over integer. Note that this type of polynomial can also be applied in other RSA-related problems, such as factoring with known bits problem [11].

In 2003, May [11] generalized the strategy by using a univariate linear polynomial to an arbitrary monic modular polynomial of degree δ, i.e. $f(x) = x^\delta + a_{\delta-1}x^{\delta-1} + \ldots + a_0 \bmod p$ where $\delta \geq 1$. As an important application, this algorithm can be used to solve the problem of factoring with known bits on Takagi's moduli $N = p^r q$ $(r > 1)$ [19].

On the other hand, in Asiacrypt'08, Herrmann and May [6] extended the univariate linear modular polynomial to polynomials with an arbitrary number of n variables. They presented a polynomial-time algorithm to find small roots of linear modular-polynomials $f(x_1, \ldots, x_n) = a_0 + a_1x_1 + \cdots + a_nx_n \bmod p$, where p is unknown and divides the known modulus N. Naturally, they applied their results to the problem of factoring with known bits for RSA modulus $N = pq$ where those unknown bits might spread across arbitrary number of blocks of p.

1.1 Our Contributions

In this paper, we focus on two variants of Herrmann-May's equations. The first is multivariate linear equations modulo an unknown divisor p^v $(v \geq 1)$ and a known composite integer N $(N \equiv 0 \bmod p^u,\ u \geq 1)$, which can be regarded as a generalization of Herrmann-May's equations $(u = 1, v = 1)$. The second is homogenous linear equations modulo an unknown divisor p^v $(v \geq 1)$ and a known composite integer N $(N \equiv 0 \bmod p^u,\ u \geq 1)$, which for $u = 1$, $v = 1$ can be seen as a special case of Herrmann-May's equations [6] $(a_0 = 0)$. Throughout this paper we suppose that $u, v \in \mathbb{Z}$.

More exactly, we investigate the problem of finding small roots of the following two classes of modular polynomials:

$$f_1(x_1, x_2, \ldots, x_n) = a_0 + a_1x_1 + \cdots + a_nx_n \bmod p^v$$
$$f_2(x_1, x_2, \ldots, x_n) = a_1x_1 + \cdots + a_nx_n \bmod p^v$$

for some unknown divisor p^v $(v \geq 1)$ and known composite integer N $(N \equiv 0 \bmod p^u,\ u \geq 1)$. We solve these two problems by introducing new techniques, which are based on, and in some sense can be viewed as a generalization of Herrmann-May's technique [6]. Applying our algorithms, we obtain better cryptanalytic results for some RSA variants, and we elaborate them below. We further conjecture that our new algorithms may find new applications in various other contexts.

Small Secret Exponent Attack on Multi-Power RSA. Here, we concentrate on an RSA variant, namely multi-power RSA, with moduli $N = p^r q$ $(r \geq 2)$. Compared to the standard RSA, the multi-power RSA is more efficient in both key generation and decryption. Besides, moduli of this type has been applied in many cryptographic designs, e.g., the Okamoto-Uchiyama cryptosystem [15], or better known via EPOC and ESIGN [4], which uses the modulus $N = p^2q$.

Suppose that the public key is (N, e), where $N = p^r q$ for some fixed $r \geq 2$ and p, q are of the same bit-size. The secret key d satisfies $ed \equiv 1 \bmod \phi(N)$, where $\phi(N)$ is Euler's ϕ-function. In Crypto'99, Takagi [19] showed that when the secret exponent $d \leq N^{\frac{1}{2(r+1)}}$, one can factorize N. Later in PKC'04, May [12] improved Takagi's bound to

$$N^{\max\{\frac{r}{(r+1)^2}, \frac{(r-1)^2}{(r+1)^2}\}}$$

In this paper, we further improve May's bound to $N^{\frac{r(r-1)}{(r+1)^2}}$, which is better than May's result when $r > 2$, and is also independent of the value of public exponent e. Similar as [12], our result

also directly implies an improved partial key exposure attack for secret exponent d with known most significant bits (MSBs) or least significant bits (LSBs). Our improvements are based on a technique for solving our first variant of Herrmann-May's equations, with the observation that $\gcd(ed-1, N) = p^{r-1}$ but $N \equiv 0 \bmod p^r$.

Weak Encryption Exponents of RSA and CRT-RSA. In Africacrypt'12, Nitaj [14] presented some attacks on RSA and CRT-RSA (the public exponent e and the private CRT-exponents d_p and d_q satisfy $ed_p \equiv 1 \bmod (p-1)$ and $ed_q \equiv 1 \bmod (q-1)$). His attacks are based on Herrmann-May's technique [6] for finding small solutions of modular equations. In particular, he reduced his attacks to solving bivariate linear modular equations modulo unknown divisors: $ex + y \equiv 0 \bmod p$ for some unknown p that divides the known modulus N.

Noticing that his equations are homogenous, we can actually improve his results with the technique for solving our second variant of Herrmann-May's equations. Besides, we extend our results to modulus $N = p^r q$.

Experimental Results. For all these attacks, we carry out experiments to verify the validity of our algorithms. The results show that our attacks are effective.

2 Preliminary

In this section, we review some useful results.

Lemma 1 (LLL [10]). *Let $\mathcal{L}$ be a lattice of dimension w. Within polynomial-time, LLL-algorithm outputs a set of reduced basis vectors v_i, $1 \leqslant i \leqslant w$ that satisfies*

$$||v_1|| \leqslant ||v_2|| \leqslant \cdots \leqslant ||v_i|| \leqslant 2^{\frac{w(w-1)}{4(w+1-i)}} \det(\mathcal{L})^{\frac{1}{w+1-i}}$$

We also state a useful lemma from Howgrave-Graham [7]. Let $g(x_1, \cdots, x_k) = \sum_{i_1,\cdots,i_k} a_{i_1,\cdots,i_k} x_1^{i_1} \cdots x_k^{i_k}$. We define the norm of g by the Euclidean norm of its coefficient vector: $||g||^2 = \sum_{i_1,\cdots,i_k} a_{i_1,\cdots,i_k}^2$.

Lemma 2 (Howgrave-Graham [7]). *Let $g(x_1, \cdots, x_k) \in \mathbb{Z}[x_1, \cdots, x_k]$ be an integer polynomial that consists of at most w monomials. Suppose that*

1. *$g(y_1, \cdots, y_k) = 0 \bmod p^m$ for $| y_1 | \leqslant X_1, \cdots, | y_k | \leqslant X_k$ and*
2. *$||g(x_1X_1, \cdots, x_kX_k)|| < \frac{p^m}{\sqrt{w}}$*

Then $g(y_1, \cdots, y_k) = 0$ holds over integers.

3 The First Variant: Generalized Linear Equations

In this section, we address how to solve $f_1(x) = a_0 + a_1 x \bmod p^v$ $(v \geq 1)$ for some unknown p where p^u divides a known modulus N (i.e. $N \equiv 0 \bmod p^u$, $u \geq 1$). In particular, the result in [8] can be viewed as a special case of our algorithm when $u = 1$, $v = 1$.

3.1 Our Result

Theorem 1. *Let N be a sufficiently large composite integer (of unknown factorization) with a divisor p^u ($p \geq N^\beta$, $u \geq 1$). Let $f_1(x) \in \mathbb{Z}[x]$ be a univariate linear polynomial. Then one can find all the solutions y of the equation $f_1(x) = 0 \pmod{p^v}$ with $v \geq 1$, $|y| \leq N^\gamma$ if*

$$\gamma < uv\beta^2$$

The time complexity of the algorithm is polynomial in $\log N$.

Proof. Consider the following univariate linear polynomial:

$$f_1(x) = a_0 + a_1 x \bmod p^v$$

where N is known to be a multiple of p^u for known u and unknown p. Here we assume that $a_1 = 1$, since otherwise we can multiply f_1 by $a_1^{-1} \bmod N$. If this inverse does not exist, one can factorize N. Let

$$f(x) = a_1^{-1} f_1(x) \bmod N$$

Define a collection of polynomials as follows:

$$g_k(x) := f^k(x) N^{\max\{\lceil \frac{v(t-k)}{u} \rceil, 0\}}$$

for $k = 0, \ldots, m$ and integer parameters t and m with $t = \tau m$ ($0 \leq \tau < 1$), which will be optimized later. Note that for all k, $g_k(y) \equiv 0 \bmod p^{vt}$.

Let $X (X = N^\gamma)$ be the upper bound on the desired root y. We built a lattice $\mathcal{L}$ of dimension $d = m + 1$ using the coefficient vectors of $g_k(xX)$ as basis vectors. We sort the polynomials according to the ascending order of g, i.e., $g_k < g_l$ if $k < l$.

From the triangular matrix of the lattice basis, we can easily compute the determinant as the product of the entries on the diagonal as $\det(\mathcal{L}) = X^s N^{s_N}$ where

$$s = \sum_{k=0}^{m} k = \frac{m(m+1)}{2} = \frac{m^2}{2} + o(m^2)$$
$$s_N = \sum_{k=0}^{t-1} \lceil \frac{v(t-k)}{u} \rceil = \frac{v\tau^2 m^2}{2u} + o(m^2)$$

To obtain a polynomial with short coefficients that contains all small roots over integer, we apply LLL basis reduction algorithm to the lattice $\mathcal{L}$. Lemma 1 gives us an upper bound on the norm of the shortest vector in the LLL-reduced basis, if the bound is smaller than the bound given in Lemma 2, we can obtain the desired polynomial. We require the following condition:

$$2^{\frac{d-1}{4}} \det(\mathcal{L})^{\frac{1}{d}} < \frac{N^{v\beta\tau m}}{\sqrt{d}}$$

where $d = m + 1$. We plug in the value for $\det(\mathcal{L})$ and d, and obtain the inequality:

$$X < 2^{-\frac{1}{2}} (m+1)^{-\frac{1}{m}} N^{2v\beta\tau - \frac{v\tau^2}{u}}$$

Neglecting the quantities that do not depend on N, and setting $\tau = u\beta$, we can get the final result

$$\gamma < uv\beta^2$$

Since our method modified [8]'s method by using a more ingenious method to choose the exponents of N, thus the parameters γ, β are independent to the dimension of the constructed lattice. The complexity of our method is dominated by LLL-algorithm, which is polynomial in $\log N$.

Eventually, the vector output by LLL-algorithm gives a univariate polynomial $g(x)$ such that $g(y) = 0$, then one can find the root of $g(x)$ over the integers. □

Extension to Arbitrary Degree. We can generalize the result of Theorem 1 to univariate polynomials of arbitrary degree.

Theorem 2. *Let N be a sufficiently large composite integer (of unknown factorization) with a divisor p^u ($p \geq N^\beta$, $u \geq 1$). Let $f_1(x) \in \mathbb{Z}[x]$ be a univariate linear polynomial of degree δ. Then one can find all the solutions y of the equation $f_1(x) = 0 \pmod{p^v}$ with $v \geq 1$, $|y| \leq N^\gamma$ if*

$$\gamma < \frac{uv\beta^2}{\delta}$$

The time complexity of the algorithm is polynomial in $\log N$.

In the proof of Theorem 2, we use the following collection of polynomials:

$$g_k(x) := x^j f^k(x) N^{\max\{\lceil \frac{v(t-k)}{u} \rceil, 0\}}$$

for $k = 0, \ldots, m$, $j = 0, \ldots, \delta - 1$ and integer parameters t and m with $t = \tau m$ ($0 \leq \tau < 1$). The rest of the proof is the same as Theorem 1. We omit it here.

Specifically, the result in [13] can be viewed as a special case of our algorithm when $u = v$.

Extension to More Variables. We can generalize the result of Theorem 1 from univariate linear equations to an arbitrary number of n variables $x_1, \ldots, x_n$ ($n \geq 2$).

Theorem 3. *Let N be a sufficiently large composite integer (of unknown factorization) with a divisor p^u ($p \geq N^\beta$, $u \geq 1$). Furthermore, let $f_1(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$ be a monic linear polynomial in $n (n \geq 2)$ variables. Under Assumption 1, we can find all the solutions $(y_1, \ldots, y_n)$ of the equation $f_1(x_1, \ldots, x_n) = 0 \pmod{p^v}$ with $v \geq 1$, $|y_1| \leq N^{\gamma_1}, \ldots |y_n| \leq N^{\gamma_n}$ if*

$$\sum_{i=1}^{n} \gamma_i < \frac{v}{u}\left(1 - (1 - \frac{u}{v}\beta)^{\frac{n+1}{n}} - (n+1)(1 - \frac{u}{v}\beta)\left(1 - \sqrt[n]{1 - \frac{u}{v}\beta}\right)\right)$$

The running time of the algorithm is polynomial in $\log N$ *but exponential in* n.

Proof. We define the following collection of polynomials which share a common root modulo p^t

$$g_{i_2,\ldots,i_n,k} = x_2^{i_2} \cdots x_n^{i_n} f_1^k N^{\max\{\lceil \frac{v(t-k)}{u} \rceil, 0\}}$$

where $i_j \in \{0, \ldots, m\}$ such that $\sum_{j=2}^{n} i_j \leq m-k$, and the parameter $t = \tau m$ has to be optimized. The idea behind the above transformation is that we try to eliminate powers of N in the diagonal entries in order to keep the lattice determinant as small as possible.

Next we can construct the lattice $\mathcal{L}$ using the similar method of Herrmann-May, therefore, the lattice has triangular form, then the determinant $\det(\mathcal{L})$ is then simply the product of the entries on the diagonal:

$$\det(\mathcal{L}) = \prod_{i=1}^{n} X_i^{s_{x_i}} N^{s_N}$$

Let d denote the dimension of $\mathcal{L}$, $t = r \cdot h + c$ ($h, c \in \mathbb{Z}$ and $0 \leq c < r$). A straightforward but tedious computation yields that

$$\begin{aligned}
s_{x_i} &= \binom{m+n}{m-1} = \frac{1}{(n+1)!} m^{n+1} + o(m^{n+1}) \\
s_N &= \sum_{k=0}^{t-1} \sum_{0 \leq \sum_{j=2}^{n} i_j \leq m-k} \lceil \frac{v(t-k)}{u} \rceil \\
&= \frac{v}{u} \frac{(n+1)\tau - 1 + (1-\tau)^{n+1}}{(n+1)!} m^{n+1} + o(m^{n+1}) \\
d &= \binom{m+n}{m} = \frac{1}{n!} m^n + o(m^n)
\end{aligned}$$

By ignoring the low-order terms, the necessary condition to obtain n equations over integer from Lemma 2 is given by

$$\det(\mathcal{L})^{\frac{1}{d-n+1}} < N^{\beta\tau m}$$

Let $X_i = N^{\gamma_i} (1 \leq i \leq n)$. Combining the values with the above condition, we obtain

$$\sum_{i=1}^{n} \gamma_i < \frac{v}{u} \left(1 - (1-\tau)^{n+1}\right) - \tau(n+1)(\frac{v}{u} - \beta)$$

By setting $\tau = 1 - \sqrt[n]{1 - \frac{u}{v}\beta}$, the condition reduces to

$$\sum_{i=1}^{n} \gamma_i < \frac{v}{u} \left(1 - (1 - \frac{u}{v}\beta)^{\frac{n+1}{n}} - (n+1)(1 - \frac{u}{v}\beta) \left(1 - \sqrt[n]{1 - \frac{u}{v}\beta}\right)\right)$$

The running time is dominated by LLL reduction, therefore, the same as the first approach, the total running time for this approach is polynomial in $\log N$ but exponential in n.

Additionally our attack relies on a well-known assumption which was widely used in the literature [5,1,6].

Assumption 1 *The lattice-based construction yields algebraically independent polynomials. The common roots of these polynomials can be efficiently computed using the* Gröbner basis technique. □

3.2 Analysis of Multi-Power RSA

We consider some multi-power RSA schemes with moduli $N = p^r q$ for $r \geq 2$, especially, two variants of RSA. In the first variant $ed \equiv 1 \bmod p^{r-1}(p-1)(q-1)$, while in the second variant $ed \equiv 1 \bmod (p-1)(q-1)$. In this section, we focus on the first variant. In Crypto'99, Takagi

Table 1. Comparisons of May's Bound, Sarkar's bound and Ours on δ

r	2	3	4	5	6	7	8	9
May's bound	0.22	0.25	0.36	0.44	0.51	0.56	0.60	0.64
Sarkar's bound	**0.39**	**0.41**	0.43	0.46	0.48	0.51	0.53	0.54
Our bound	0.22	0.37	**0.48**	**0.55**	**0.61**	**0.65**	**0.69**	**0.72**

[19] proved that when the decryption exponent $d < N^{\frac{1}{2(r+1)}}$, one can factorize N in polynomial time. Later, in PKC'04, May [12] improved Takagi's bound to

$$N^{\max\{\frac{r}{(r+1)^2}, \frac{(r-1)^2}{(r+1)^2}\}}$$

Based on the technique of Theorem 1, we can further improve May's bound to $N^{\frac{r(r-1)}{(r+1)^2}}$.

Theorem 4. *Let $N = p^r q$, where $r \geq 2$ is a known integer and p, q are primes of the same bit-size. Let e be the public key exponent and d be the private key exponent, satisfying $ed \equiv 1 \bmod \phi(N)$. Suppose that*

$$d < N^{\frac{r(r-1)}{(r+1)^2}}$$

Then N can be factored in polynomial time.

Proof. Since $\phi(N) = p^{r-1}(p-1)(q-1)$, we have the following equation

$$ed - 1 = kp^{r-1}(p-1)(q-1) \quad \text{for some } k \in \mathbb{N}$$

Then we want to find the root $y = d$ of the polynomial

$$f_1(x) = ex - 1 \bmod p^{r-1}$$

with the known multiple (of unknown divisor p) N ($N \equiv 0 \bmod p^r$). Let $d \approx N^{\delta}$. Applying Theorem 1, setting $\beta = \frac{1}{r+1}$, $u = r$, $v = r-1$, we obtain the final result

$$\delta < \frac{r(r-1)}{(r+1)^2}$$

□

Recently, Sarkar [17] improved May's bound for modulus $N = p^r q$, however, his method can not applied for any public key exponents e of arbitrary size. In addition, we get better experimental results for the case of $r > 2$.

For small r, we provide the comparison of May's bound, Sarkar's bound, and our bound on δ in Table 1. Note that for $r = 2$, we obtain the same result as May's bound.

Similar to the results of [12], the new attack of Theorem 4 immediately implies partial key exposure attacks for d with known MSBs/LSBs. Following we extend the approach of Theorem 4 to partial key exposure attacks.

Theorem 5 (MSBs). *Let $N = p^r q$, where $r \geq 2$ is a known integer and p, q are primes of the same bit-size. Let e be the public key exponent and d be the private key exponent, satisfying $ed = 1 \bmod \phi(N)$. Given $\tilde{d}$ such that*

$$|d - \tilde{d}| < N^{\frac{r(r-1)}{(r+1)^2}}$$

Then N can be factored in polynomial time.

Table 2. Experimental Results of the Attack from Theorem 4

N (bit)	r	e	d-pred(bits)	(m,t)	dim($\mathcal{L}$)	d-exp(bits)	time(sec)
1536	2	1536	341	$(30,20)$	31	315	2354
2048	3	2048	768	$(20,15)$	21	700	671
2048	3	4096	768	$(20,15)$	21	700	711
2048	3	2048	768	$(40,30)$	41	735	29228
2560	4	2560	1228	$(20,16)$	21	1135	628
2560	4	2560	1228	$(30,24)$	31	1165	9159

Proof. We have that

$$e(d-\tilde{d})+e\tilde{d}-1 \equiv 0 \bmod p^{r-1}$$

Then we want to find the root $y = d - \tilde{d}$ of the polynomial

$$f_1(x) = ex + e\tilde{d} - 1 \bmod p^{r-1}$$

with the known multiple (of unknown divisor p) N ($N \equiv 0 \bmod p^r$). Applying Theorem 1, setting $\beta = \frac{1}{r+1}$, $u = r$, $v = r-1$, we obtain the final result. □

Theorem 6 (LSBs). *Let $N = p^r q$, where $r \geq 2$ is a known integer and p, q are primes of the same bit-size. Let e be the public key exponent and d be the private key exponent, satisfying $ed = 1 \bmod \phi(N)$. Given d_0, M with $d = d_0 \mod M$ and*

$$M > N^{\frac{3r+1}{(r+1)^2}}$$

Then N can be factored in polynomial time.

Proof. Rewrite $d = d_1 M + d_0$, then we have

$$ed_1M + ed_0 - 1 \equiv 0 \bmod p^{r-1}$$

Then we want to find the root $y = d_1$ of the polynomial

$$f_1(x) = eMx + ed_0 - 1 \bmod p^{r-1}$$

with the known multiple (of unknown divisor p) N ($N \equiv 0 \bmod p^r$). Applying Theorem 1 and setting $\beta = \frac{1}{r+1}$, $u = r$, $v = r-1$, we obtain the final result. □

3.3 Experimental Results

We have implemented the attack of Section 3.2 using Magma [21] on a laptop with Intel© Core™ i5-2430M CPU 2.40 GHz, 2 GB RAM. Table 2 shows the experimental results for multi-power RSA modulus N with 512-bit primes p, q.

We compute the number of bits that one should theoretically be able to attack for d_p (column d_p-pred in Table 2). In all the listed experiments, we can recover the factorization of N. Note that our attack is independent of the value of public exponent e.

4 The Second Variant: Homogenous Linear Equations

In this section, we study the problem of finding small roots of homogenous linear polynomials $f_2(x_1, x_2) = a_1x_1 + a_2x_2 \bmod p^v$ $(v \geq 1)$ for some unknown p where p^u divides a known modulus N (i.e. $N \equiv 0 \bmod p^u$, $u \geq 1$). Let (y_1, y_2) be a small solution of $f_2(x_1, x_2)$. We assume that we also know an upper bound $(X_1, X_2) \in \mathbb{Z}^2$ for the root such that $|y_1| \leq X_1, |y_2| \leq X_2$.

4.1 Our Result

Theorem 7. *Let N be a sufficiently large composite integer (of unknown factorization) with a divisor p^u $(p \geq N^\beta$, $u \geq 1)$. Let $f_2(x_1, x_2) \in \mathbb{Z}[x_1, x_2]$ be a homogenous linear polynomial in two variables. Then one can find all the solutions (y_1, y_2) of the equation $f_2(x_1, x_2) = 0 \pmod{p^v}$ $(v \geq 1)$ with $\gcd(y_1, y_2) = 1$, $|y_1| \leq N^{\gamma_1}, |y_2| \leq N^{\gamma_2}$ if*

$$\gamma_1 + \gamma_2 < uv\beta^2$$

The time complexity of the algorithm is polynomial in $\log N$.

Proof. Since the proof is similar to that of Theorem 1, we only give the sketch here. Consider the linear polynomial:

$$f_2(x_1, x_2) = a_1x_1 + a_2x_2 \bmod p^v$$

where N is known to be a multiple of p^u for known u and unknown p. Here we assume that $a_1 = 1$, since otherwise we can multiply f_2 by $a_1^{-1} \bmod N$. If this inverse does not exist, one can factorize N. Let

$$f(x_1, x_2) = a_1^{-1} f_2(x_1, x_2) \bmod N$$

Define a collection of polynomials as follows:

$$g_k(x_1, x_2) := x_2^{m-k} f^k(x_1, x_2) N^{\max\{\lceil \frac{v(t-k)}{u} \rceil, 0\}}$$

for $k = 0, \ldots, m$ and integer parameters t and m with $t = \tau m$ $(0 \leq \tau < 1)$, which will be optimized later. Note that for all k, $g_k(y_1, y_2) \equiv 0 \bmod p^{vt}$. Let $X_1, X_2 (X_1 = N^{\gamma_1}, X_2 = N^{\gamma_2})$ be upper bounds on the desired root (y_1, y_2). We build a lattice $\mathcal{L}$ of dimension $d = m + 1$ using the coefficient vectors of $g_k(x_1X_1, x_2X_2)$ as basis vectors. We sort the polynomials according to the order as following: If $k < l$, then $g_k < g_l$.

From the triangular matrix of the lattice, we can easily compute the determinant as the product of the entries on the diagonal as $\det(\mathcal{L}) = X_1^{s_1} X_2^{s_2} N^{s_N}$ where

$$s_1 = s_2 = \sum_{k=0}^{m} k = \frac{m(m+1)}{2} = \frac{m^2}{2} + o(m^2)$$

$$s_N = \sum_{k=0}^{t-1} \lceil \frac{v(t-k)}{u} \rceil = \frac{vt^2}{2u} = \frac{v\tau^2 m^2}{2u} + o(m^2)$$

Combining Lemma 2 and Lemma 1, after some calculations, we can get the final result

$$\gamma_1 + \gamma_2 \leq uv\beta^2$$

The complexity of this method is polynomial in $\log N$.

The vector output by LLL-algorithm gives a polynomial $f^{'}(x_1, x_2)$ such that $f^{'}(y_1, y_2) = 0$. Next we try to extract the secret root: According to Bézot's theorem, the irreducible polynomial $h(x_1, x_2) = y_1x_2 - y_2x_1$ must divide $f^{'}(x_1, x_2)$. Therefore, we can obtain an integer multiple $b \cdot h(x_1, x_2) = h_1x_1 + h_2x_2$ of $h(x_1, x_2)$ by factoring $f^{'}(x_1, x_2)$ into irreducible polynomials over $\mathbb{Q}(x_1, x_2)$. Since $\gcd(y_1, y_2) = 1$, we obtain $y_1 = \frac{h_1}{\gcd(h_1,h_2)}$ and $y_2 = \frac{h_2}{\gcd(h_1,h_2)}$. □

Comparisons with Previous Methods. For $u = 1, v = 1$, the upper bound $\delta_1+\delta_2$ of Theorem 7 is β^2, that is exactly May's results [11] on univariate linear polynomial $f(x) = x + a$. Actually the problem of finding a small root of homogenous polynomial $f(x_1, x_2)$ can be transformed to find small rational roots of univariate linear polynomial $F(z)$ i.e. $F(\frac{x_2}{x_1}) = f(x_1, x_2)/x_1$ (the discussions of the small rational roots can be found in Joux's book [9]).

Our result improves Herrmann-May's bound $3\beta - 2 + 2(1 - \beta)^{\frac{3}{2}}$ up to β^2 if $a_0 = 0$. As a concrete example, for the case $\beta = 0.5$, our method improves the upper size of X_1X_2 from $N^{0.207}$ to $N^{0.25}$.

Another important work to mention is that in [2], Castagnos, Joux, Laguillaumie and Nguyen also considered homogenerous polynomials. Their algorithm can be directly applied to our attack scenario. Consider the following bivariate homogeneous polynomial

$$f(x_1, x_2) = (a_1x_1 + a_2x_2)^{\frac{u}{v}} \bmod p$$

Therefore, their algorithm can only deal with the cases $\frac{u}{v} \in \mathbb{Z}$, and our algorithm is simpler and more effective, specially, for $\frac{u}{v}$-degree polynomial with $2^{\frac{u}{v}}$ monomials (the dimension of lattice is $\frac{u}{v}m$), whereas our algorithm is for linear polynomial with two monomials (the dimension of lattice is m). Besides, in [2], they formed a lattice using the coefficients of $g(x, y)$ instead of $g(xX, yY)$. This modification enjoys the benefits in terms of real efficiency, since their lattice has smaller determinant than in the classical bivariate approach. However, their algorithm fails when the solutions are significantly unbalanced ($X_1 \gg X_2$). We highlight the idea that the factor X, Y should not only be used to balance the size of different power of x, y but also to balance the variables x, y. That is why our algorithm is suitable for this unbalanced attack scenario.

Extension to More Variables. We generalize the result of Theorem 7 from bivariate linear equations to an arbitrary number of n variables $x_1, \ldots, x_n$. The following result is similar to Theorem 3, we only state here.

Theorem 8. *Let N be a sufficiently large composite integer (of unknown factorization) with a divisor p^u ($p \geq N^\beta$, $u \geq 1$). Furthermore, let $f_2(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$ be a homogenous linear polynomial in $n (n \geq 3)$ variables. Under Assumption 1, we can find all the solutions $(y_1, \ldots, y_n)$ of the equation $f_2(x_1, \ldots, x_n) = 0 \pmod{p^v}$ ($v \geq 1$) with $\gcd(y_1, \ldots, y_n) = 1$, $|y_1| \leq N^{\gamma_1}, \ldots |y_n| \leq N^{\gamma_n}$ if*

$$\sum_{i=1}^{n} \gamma_i < \frac{v}{u}\left(1 - (1 - \frac{v}{u}\beta)^{\frac{n}{n-1}} - n(1 - \frac{v}{u}\beta)\left(1 - \sqrt[n-1]{1 - \frac{v}{u}\beta}\right)\right)$$

The running time of the algorithm is polynomial in $\log N$ but exponential in n.

4.2 Applications

In Africacrypt'12 [14], Nitaj presented a new attack on RSA. His attack is based on Herrmann-May's method [6] for finding small roots of a bivariate linear equation. In particular, he showed

that the public modulus N can be factored in polynomial-time for the RSA cryptosystem where the public exponent e satisfies an equation $ex+y \equiv 0 \pmod{p}$ with parameters x and y satisfying $ex + y \not\equiv 0 \pmod{N}$ $|x| < N^{\gamma}$ and $|y| < N^{\delta}$ with $\delta + \gamma \leq \frac{\sqrt{2}-1}{2}$.

Note that the equation of [14] is homogenous, thus we can improve the upper bound of $\gamma + \delta$ using our idea as Theorem 7. In [16], Sarkar proposed another method to extend Nitaj's weak encryption exponents, the trick is to consider the fact that Nitaj's bound can be improved when the unknown variables in the modular equation are unbalanced (x and y are of different bit-size). In general, Sarkar's method is essentially Herrmann-May's method, whereas our algorithm is simpler (see Theorem 7). We present our result below.

Theorem 9. *Let $N = pq$ be an RSA modulus with $q < p < 2q$. Let e be a public exponent satisfying an equation $ex + y \equiv 0 \pmod{p}$ with $|x| < N^{\gamma}$ and $|y| < N^{\delta}$. If $ex + y \not\equiv 0 \pmod{N}$ and $\gamma + \delta \leq 0.25$, N can be factored in polynomial-time.*

In [14], Nitaj also proposed a new attack on CRT-RSA. Let $N = pq$ be an RSA modulus with $q < p < 2q$. Nitaj showed that if $e < N^{\frac{\sqrt{2}}{2}}$ and $ed_p = 1 + k_p(p-1)$ for some d_p with $d_p < \frac{N^{\frac{\sqrt{2}}{4}}}{\sqrt{e}}$, N can be factored in polynomial-time. His method is also based on Herrmann-May's method. Similarly we can improve Nitaj's result using our idea as Theorem 7.

Theorem 10. *Let $N = pq$ be an RSA modulus with $q < p < 2q$. Let e be a public exponent satisfying $e < N^{0.75}$ and $ed_p = 1 + k_p(p-1)$ for some d_p with*

$$d_p < \frac{N^{0.375}}{\sqrt{e}}$$

Then, N can be factored in polynomial-time.

Proof. We rewrite the equation $ed_p = 1 + k_p(p-1)$ as

$$ed_p + k_p - 1 = k_p p$$

Then we focus on the equation modulo p

$$ex + y = 0 \bmod p$$

with a root $(x_0, y_0) = (d_p, k_p - 1)$. Suppose that $e = N^{\alpha}$, $d_p = N^{\delta}$, then we get

$$k_p = \frac{ed_p - 1}{p-1} < \frac{ed_p}{p-1} < N^{\alpha+\delta-0.5}$$

Applying Theorem 7 with the desired equation where $x_0 = d_p < N^{\delta}$ and $y_0 = k_p - 1 < N^{\alpha+\delta-0.5}$, setting $\beta = 0.5$, $u = 1$ and $v = 1$ we obtain

$$2\delta + \alpha < 0.75$$

Note that $\gcd(x_0, y_0) = (d_p, k_p - 1) = 1$, $k_p < N^{\alpha+\delta-0.5} < N^{\alpha+2\delta-0.5} < N^{0.25} < p$, hence $ed_p + k_p - 1 \neq 0 \bmod N$. Then we can factorize N with $\gcd(N, ed_p + k_p - 1) = p$. □

Moreover, we extend small secret exponent attack on Takagi's scheme [19] with moduli $N = p^r q$ $(r \geq 2)$.

Table 3. Experimental Results for Weak Encryption Exponents

N (bit)	r	d_p-pred(bits)	(m,t)	dim($\mathcal{L}$)	d_p-exp(bits)	time(sec)
1024	1	128	$(6,3)$	7	110	<1
1024	1	128	$(10,5)$	11	115	<1
1024	1	128	$(30,15)$	31	124	340
1536	2	170	$(10,6)$	11	140	1
1536	2	170	$(30,20)$	31	160	449
2048	3	192	$(10,7)$	11	135	1
2048	3	192	$(48,36)$	49	180	10584

Theorem 11. *Let $N = p^r q$ be a Takagi's RSA variant modulus. Let $e < (p-1)(q-1)$, $d_p < p-1$ be a public exponent and private CRT-exponent, satisfying $ed_p = 1 \bmod (p-1)$. Suppose that $p < N^\beta$, $e < N^\alpha$, $d_p < N^\delta$. Then N can be factored in polynomial-time provided that*

$$\delta < \frac{r\beta^2 + \beta - \alpha}{2}$$

Proof. We rewrite the equation $ed_p = 1 + k_p(p-1)$ as $ed_p + k_p - 1 = k_p p$. Then we focus on the equation modulo p

$$ex + y = 0 \bmod p$$

with a root $(x_0, y_0) = (d_p, k_p - 1)$. We have $p = N^\beta$, $e = N^\alpha$ and $d_p = N^\delta$, then we get

$$k_p = \frac{ed_p - 1}{p-1} < \frac{ed_p}{p-1} < N^{\alpha+\delta-\beta}$$

Applying Theorem 7 with the desired equation where $x_0 = d_p < N^\delta$ and $y_0 = k_p - 1 < N^{\alpha+\delta-\beta}$, setting $u = r$ and $v = 1$ we obtain

$$2\delta + \alpha < r\beta^2 + \beta$$

Note that $\gcd(x_0, y_0) = (d_p, k_p - 1) = 1$. Then we can factorize N with $\gcd(N, ed_p + k_p - 1) = p$. □

4.3 Experimental Results

Table 3 shows the experimental results for multi-power RSA modulus N with 512-bit primes p, q.

In all of our experiments, we fix e's length as 512-bit. We also compute the number bits that one should theoretically be able to attack for d_p (column d_p-pred of Table 3).

For $r = 1$, that is actually the attack described in Theorem 10. In [14], the author showed that for a 1024-bit modulus N, the CRT-exponent d_p is typically of size at most 110. We obtain better results in our experiments as shown in Table 3.

References

1. D. Boneh and G. Durfee. Cryptanalysis of RSA with private key d less than $N^{0.292}$. *IEEE Transactions on Information Theory*, 46(4):1339–1349, 2000. 6

2. G. Castagnos, A. Joux, F. Laguillaumie, and P. Nguyen. Factoring pq^2 with quadratic forms: Nice cryptanalyses. In *Asiacrypt 2009*, pages 469–486. Springer, 2009. 10
3. D. Coppersmith. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. *Journal of Cryptology*, 10(4):233–260, 1997. 1
4. The EPOC and the ESIGN Algorithms. IEEE P1363: Protocols from Other Families of Public-Key Algorithms. http://grouper.ieee.org/groups/1363/StudyGroup/NewFam.html, 1998. 2
5. M. Ernst, E. Jochemsz, A. May, and B. De Weger. Partial key exposure attacks on RSA up to full size exponents. *Advances in Cryptology–EUROCRYPT 2005*, pages 555–555, 2005. 6
6. M. Herrmann and A. May. Solving linear equations modulo divisors: On factoring given any bits. *Advances in Cryptology-ASIACRYPT 2008*, pages 406–424, 2008. 2, 3, 6, 10
7. N. Howgrave-Graham. Finding small roots of univariate modular equations revisited. *Crytography and Coding*, pages 131–142, 1997. 3
8. N. Howgrave-Graham. Approximate integer common divisors. *Cryptography and Lattices*, pages 51–66, 2001. 1, 3, 5
9. A. Joux. *Algorithmic cryptanalysis*. Chapman & Hall/CRC, 2009. 10
10. A.K. Lenstra, H.W. Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982. 1, 3
11. A. May. *New RSA vulnerabilities using lattice reduction methods*. PhD thesis, 2003. 1, 2, 10
12. A. May. Secret exponent attacks on RSA-type schemes with moduli $N = p^r q$. *Public Key Cryptography–PKC 2004*, pages 218–230, 2004. 2, 7
13. A. May. Using lll-reduction for solving RSA and factorization problems. *The LLL algorithm*, pages 315–348, 2010. 5
14. A. Nitaj. A new attack on RSA and CRT-RSA. *Progress in Cryptology-AFRICACRYPT 2012*, pages 221–233, 2012. 3, 10, 11, 12
15. T. Okamoto and S. Uchiyama. A new public-key cryptosystem as secure as factoring. *Advances in Cryptography–Eurocrypt'98*, pages 308–318, 1998. 2
16. S. Sarkar. Reduction in lossiness of RSA trapdoor permutation. In *Security, Privacy, and Applied Cryptography Engineering*, pages 144–152. Springer, 2012. 11
17. S. Sarkar. Small secret exponent attack on RSA variant with modulus $N = p^r q$. *Designs, Codes and Cryptography*, pages 1–10, 2014. 7
18. A. Shamir. A polynomial time algorithm for breaking the basic Merkle-Hellman cryptosystem. In *Foundations of Computer Science, 1982. SFCS'08. 23rd Annual Symposium on*, pages 145–152. IEEE, 1982. 1
19. T. Takagi. Fast RSA-type cryptosystem modulo $p^k q$. In *Advances in Cryptology–Crypto'98*, pages 318–326. Springer, 1998. 2, 7, 11
20. M. Van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan. Fully homomorphic encryption over the integers. In *Advances in Cryptology–EUROCRYPT 2010*, pages 24–43. Springer, 2010. 1
21. J. Cannon W. Bosma and C. Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. Computational algebra and number theory (London, 1993). 8