A New Partial Key Exposure Attack on Multi-power RSA*

Muhammed F. Esgin[1,2], Mehmet S. Kiraz[1], and Osmanbey Uzunkol[1]

[1] TÜBİTAK BİLGEM UEKAE, Turkey
{muhammed.esgin, mehmet.kiraz, osmanbey.uzunkol}@tubitak.gov.tr
[2] Graduate School of Natural and Applied Sciences, İstanbul Şehir University, Turkey

Abstract. An important attack on multi-power RSA ($N = p^r q$) was introduced by Sarkar in 2014, by extending the small private exponent attack of Boneh and Durfee on classical RSA. In particular, he showed that N can be factored efficiently for $r = 2$ with private exponent d satisfying $d < N^{0.395}$. In this paper, we generalize this work by introducing a new partial key exposure attack for finding small roots of polynomials using Coppersmith's algorithm and Gröbner basis computation. Our attack works for all multi-power RSA exponents e (resp. d) when the exponent d (resp. e) has full size bit length. The attack requires prior knowledge of least significant bits (LSBs), and has the property that the required known part of LSB becomes smaller in the size of e. For practical validation of our attack, we demonstrate several computer algebra experiments.

Keywords: Multi-power RSA, Integer factorization, Partial key exposure, Coppersmith's method, Small roots of polynomials.

1 Introduction

A natural way of speeding up the decryption/signing procedure of RSA based cryptographic schemes is to use a small private exponent d. However, Wiener [22] showed that classical RSA construction becomes insecure when $d < \frac{1}{3}N^{\frac{1}{4}}$. Later, this bound was further improved by Boneh and Durfee [2] to $N^{0.292}$ by using results of Coppersmith [6].

Kocher [15] initiated a new type of attack that obtains information about the bits of d using side-channel techniques in 1996. The idea is to exploit certain weaknesses of the actual implementation (e.g., execution time, power consumption, noise), which in turn reveals some bits of d. In general, the attacker gains information about either consecutive least significant bits (LSBs) or most significant bits (MSBs). Therefore, partial key exposure attacks mostly focus on these two rather specific cases.

* An earlier version of this paper appeared in Conference on Algebraic Informatics (CAI) 2015.

Boneh, Durfee and Frankel [3] introduced the first algebraic partial key exposure attack using partial information of d. The attack finds the whole secret exponent d when sufficient partial knowledge of d is known. Coppersmith's algorithm for finding small roots of polynomials is used in such algebraic attacks [6,5,4]. This algorithm uses lattice reduction techniques to obtain efficient small roots of certain polynomials (in particular, the LLL algorithm [16]). Later, new partial key exposure attacks on classical RSA were described by Blömer and May in [1]. We refer to [9,14] for further partial key exposure attacks on standard RSA.

Notation: Let log denote the logarithm base 2 unless the base is given concretely. We use the following notation throughout this manuscript.

N	Multi-power RSA modulus
n	bitsize of N
p, q	prime factors of N
r	integer satisfying the relation $N = p^r q$
e	RSA public exponent
d	RSA private exponent
d_0	known part of d
α	$\log_N e$ (i.e., $e \approx N^\alpha$)
β	$\log_N d$ (i.e., $d \approx N^\beta$)
δ	$\log_N d_0$ (i.e., $d_0 \approx N^\delta$)

In this work, we focus on multi-power RSA (also referred as Takagi's RSA or prime power RSA) introduced by Takagi in [21]. One of the motivation of this variant is to speed up the RSA decryption/signing process. More concretely, $N = p^r q$ is chosen for two (distinct) primes of same bit length such that $r \geq 2$. Then, there are two different ways of generating public/private exponents. The first one imposes the condition $ed \equiv 1 \bmod (p-1)(q-1)$ while the other $ed \equiv 1 \bmod \phi(N)$, where $\phi(N) = p^{r-1}(p-1)(q-1)$. Decryption of a ciphertext c is computed more efficiently using simply a combination of Hensel lifting and Chinese Remainder Theorem modulo p^r and q (see [21] for details).

For the multi-power RSA variant when exponents are generated modulo $\phi(N)$, Takagi proved in [21] that if $d \leq N^{\frac{1}{2r+2}}$, then N can be factored. This was later improved by May in [18] to $d < \max\{N^{\frac{r}{(r+1)^2}}, N^{\frac{(r-1)^2}{(r+1)^2}}\}$. Recently, Sarkar [20] improved this bound even further for $r \leq 5$ and showed in particular that if $d < N^{0.395}$ and $r = 2$, then N can be factored efficiently. Thereafter, Lu et al. [17] improved Sarkar's result for $r \geq 4$. Their attack works when the unknown part $\tilde{d}$ of d (it may be all of d or an MSB/LSB part of it) satisfies $\tilde{d} < N^{\frac{r(r-1)}{(r+1)^2}}$.

In [13], a small private exponent attack is shown for the case when exponents are generated modulo $(p-1)(q-1)$. This attack shows that N can be factored if $d < N^{\frac{2-\sqrt{2}}{r+1}}$. Later, the idea of this work is used in [12] for partial key exposure attacks. For instance, for $r = 2$ and $e \approx N^{\frac{2}{3}}$, it is shown that N can be factored

in any of the following conditions:

$$\begin{aligned} & \gamma \leq \tfrac{7}{12} - \tfrac{1}{4}\sqrt{\tfrac{16+24\beta}{3} - \tfrac{39}{9}} && \text{if MSBs or middle bits are known,} \\ \text{or} \quad & \gamma \leq \tfrac{5}{9} - \tfrac{2}{3}\sqrt{\tfrac{2+3\beta}{3} - \tfrac{5}{9}} && \text{if LSBs are known,} \end{aligned}$$

where $d \approx N^{\beta}$ and the unknown part of d is approximately N^{γ}. Note that their attacks do not work when d is of full size modulo $(p-1)(q-1)$ (i.e., $d \approx N^{\frac{2}{3}}$).

Our Contribution. In this paper, we provide a new partial key exposure attack on multi-power RSA when the exponents are generated modulo $\phi(N)$. The attack basically uses partial knowledge of LSBs and works for all e (resp. d) when the exponent d (resp. e) has full size bit length.[3] More concretely, we prove the following theorem which generalizes Sarkar's result [20].

Theorem 1. *Let $r \geq 2$ be an integer and $N = p^r q$ be a multi-power RSA modulus, where p and q are distinct primes with the same bit size (i.e., $p, q \approx N^{\frac{1}{r+1}}$). Suppose that $ed \equiv 1 \bmod \phi(N)$ with $e \approx N^{\alpha}$ and $d \approx N^{\beta}$. Suppose further that an attacker obtains an LSB part d_0 of d, where $d_0 \geq N^{\delta}$ for some $\delta \in \mathbb{R}^{\geq 0}$. Then under Assumption 1, there exists an algorithm which finds the prime factors of N in polynomial time in $\log N$ provided that*

$$\rho(r, \beta, \alpha, \delta) < 0,$$

where ρ is a function of r, β, α and δ.

We show the improvement of our attack over Sarkar's result in Figure 1 for the case $r = 2$. Light grey area (indicated by "Sarkar'14") shows the attack region by [20] and darker grey areas are the applicable regions of our attack.

Organization of the paper. In Section 2, we give preliminaries about lattices. In Section 3, we prove our main result, Theorem 1, extending the result of [20]. Section 4 demonstrates several experiments justifying our claims for the multi-power RSA moduli of length 1024 or 2048 bits. We conclude the paper in Section 5 and argue the improbability of using our attack for known MSBs by addressing an issue in [12].

2 Preliminaries

In this section, we give basic definitions and theorems about lattices. Let $v = (a_0, \cdots, a_s)$ be a vector in $\mathbb{R}^{s+1}$ for some $s \geq 0$. We use the Euclidean norm $||v||$ of v

$$||v|| := \sqrt{\sum_{i=0}^{s} (a_i)^2}.$$

[3] This rule is induced by the condition that $ed \equiv 1 \bmod \phi(N)$.

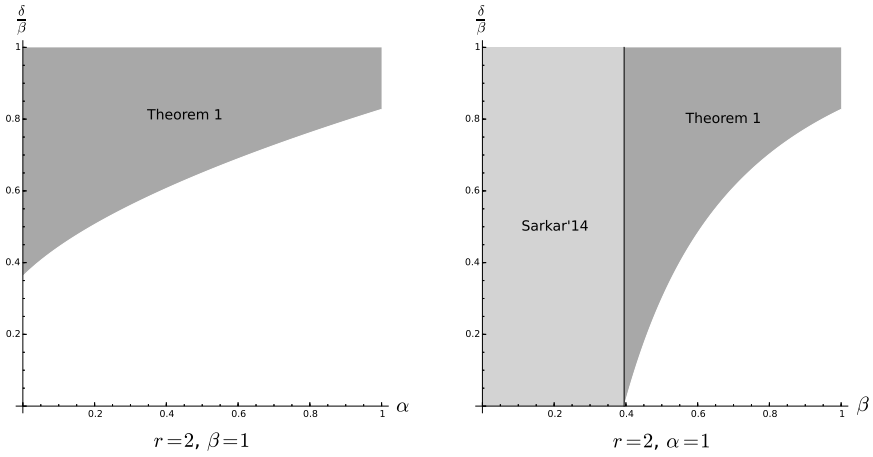

Fig. 1. The relation between the sizes of e (resp. d) and the fraction of the part of d required to be known.

For a multivariate polynomial f, the norm $||f||$ of f is the Euclidean norm of its coefficient vector. Let $v_1, \cdots, v_w \in \mathbb{R}^m$ be a set of $\mathbb{R}$-linearly independent vectors with $w, m \in \mathbb{N}^{>0}$ and $w \leq m$. Then, the lattice L generated by these vectors is

$$L := \{b_1 v_1 + \cdots + b_w v_w \ : \ b_i \in \mathbb{Z} \text{ for } 1 \leq i \leq w\}.$$

We always work on lattice having *full rank*, i.e. $w = m$. We denote $dim(L) := w$ for the dimension of L. Each lattice L can be represented by the following matrix $\mathcal{M} \in \mathrm{GL}(w, \mathbb{R})$:

$$\mathcal{M} = \begin{pmatrix} v_1 \\ . \\ . \\ . \\ v_w \end{pmatrix}.$$

We denote $det(L)$ for the determinant of L. We have $det(L) = det(\mathcal{M})$ for a full rank lattice L.

In this work, the main goal is to find small vectors in such full lattices. Computational complexity of finding the smallest vector in a lattice increases exponentially in $dim(L)$. The reduction algorithm LLL introduced by Lenstra, Lenstra and Lovász [16] is generally used in practice to have an efficient lattice reduction technique for obtaining small enough basis vectors. The following theorem gives an upper bound on the norm of the reduced basis vectors output by the LLL algorithm.

Theorem 2. *Let L be a lattice with $dim(L) = w$ as above. The LLL algorithm produces a set of reduced basis vectors $\{R_1, \cdots, R_w\}$ such that*

$$||R_i|| \leq 2^{\frac{w(w-1)}{4(w+1-i)}} det(L)^{\frac{1}{w+1-i}}.$$

The computational complexity of the LLL algorithm is polynomial in $dim(L)$ and in the maximal bitsize of an entry [19].

Coppersmith described methods for finding small roots of univariate and bivariate polynomials [4,5,6]. The methods can be extended to the polynomials having more variables, but the results become heuristic. Howgrave-Graham [11] reformulated these results and proved the following theorem:

Theorem 3 (Howgrave-Graham's Theorem, [11]).
Let $f(x_1, \cdots, x_s) \in \mathbb{Z}[x_1, \cdots, x_s]$ be a polynomial for $s \geq 1$. Assume that the number of monomials is less than or equal to w. If the following two conditions hold:

1. *$M \in \mathbb{Z}^+$ and $f(x_1^0, \cdots, x_s^0) \equiv 0 \mod M$ for some $|x_1^0| < X_1, \cdots, |x_s^0| < X_s$,*
2. *$||f(x_1X_1, \cdots, x_sX_s)|| < \frac{M}{\sqrt{w}}$,*

then $(x_1^0, \cdots, x_s^0)$ is a root of f over $\mathbb{Z}$.

After finding multivariate polynomials carrying a common root over integers, we need to extract this root using Gröbner basis computation.[4] Our main result Theorem 1 is valid under the following assumption:

Assumption 1 *Let $f_1, \cdots, f_k$ be the polynomials having the desired root over $\mathbb{Z}$ for $k \geq 3$ computed using LLL reduction. Furthermore, let I be the ideal generated by these polynomials. Then, the algebraic variety of I is zero-dimensional. In particular, the common root can be extracted by computing a Gröbner basis on I.*

Since our result in Theorem 1 relies on this assumption, it is heuristic. However, our experiments show that this assumption holds in general (see Section 4). The computational complexity of a Gröbner basis computation can be bounded by a polynomial in $\log N$ assuming the number of variables and the maximal degree of input polynomials is fixed [10].

3 An Attack with Known LSBs

In this section, we prove our main Theorem 1.

Proof (Theorem 1). Multi-power RSA parameters satisfy the congruence $ed \equiv 1 \mod \phi(N)$ with $\phi(N) = (p^r - p^{r-1})(q-1)$. This implies the equation that $ed - 1 = k(p^r - p^{r-1})(q-1)$ for some $k \in \mathbb{Z}$. Since we know an LSB part of d,

[4] Resultant computation could be another option as well, but it was less efficient for our experiments.

we can write this as $eM\tilde{d} + ed_0 - 1 = k(p^r - p^{r-1})(q-1)$ where $d = \tilde{d}M + d_0$ and M is a power of 2. Hence, we have the following polynomial

$$f_{eM}(x, y, z) = ed_0 - 1 - xN - xy^{r-1} + xy^{r-1}z + xy^r$$

carrying the root $(x_0, y_0, z_0) = (k, p, q)$ modulo eM. It is easy to see that $|x_0| < X := N^{\alpha+\beta-1}$, $|y_0| < Y := N^{\frac{1}{r+1}}$ and $|z_0| < Z := N^{\frac{1}{r+1}}$ neglecting small constants.

Let $m, t_1, t_2 \geq 0$ and define the following shift polynomials:

$$\begin{aligned}
g_{i,j,k}(x, y, z) &= x^j y^k z^{j+t_1} f_{eM}^i(x, y, z), \\
&\qquad \text{where } i = 0, \cdots, m,\ j = 1, \cdots, m-i \text{ and } k = j, \cdots, j+2r-2, \\
g_{i,0,k}(x, y, z) &= y^k z^{t_1} f_{eM}^i(x, y, z), \\
&\qquad \text{where } i = 0, \cdots, m \text{ and } k = 0, \cdots, t_2.
\end{aligned}$$

Recall that $y_0^r z_0 = N$. Hence, we replace every occurrence of $y^r z$ with N in the shift polynomials. Denote new polynomials by $g'_{i,j,k}(x, y, z)$. Observe that choosing xy^r as the leading monomial of f_{eM}, the leading monomials in $g'_{i,j,k}$'s are of the form $x^{i+j} y^{k+ri-rl} z^{j+t_1-l}$, where $l = \min\left\{\lfloor \frac{k+ri}{r} \rfloor, j + t_1\right\}$.

Algorithm 1 Generating the Lattice L

Input: $r \geq 2$; $m, t_1, t_2 \geq 0$ and $f_{eM}(x, y, z)$

$G, H, Ord \leftarrow \emptyset$
for $i \in \{0, 1, \cdots, m\}$ **do**
  for $j \in \{1, 2, \cdots, m-i\}$ **do**
    for $k \in \{j, j+1, \cdots, j+2r-2\}$ **do**
      Append $(x^j y^k z^{j+t_1} f_{eM}^i, i)$ to G
      $l \leftarrow \min\left\{\lfloor \frac{k+ri}{r} \rfloor, j + t_1\right\}$
      Append $x^{i+j} y^{k+ri-rl} z^{j+t_1-l}$ to Ord
    end for
  end for
end for
for $i \in \{0, 1, \cdots, m\}$ **do**
  for $k \in \{0, 1, \cdots, t_2\}$ **do**
    Append $(y^k z^{j+t_1} f_{eM}^i, i)$ to G
    $l \leftarrow \min\left\{\lfloor \frac{k+ri}{r} \rfloor, j + t_1\right\}$
    Append $x^{i+j} y^{k+ri-rl} z^{j+t_1-l}$ to Ord
  end for
end for
for each element (g, i) in G **do**
  Replace each occurrence of $y^r z$ with N in g
  $a'_\ell \leftarrow a_\ell^{-1} \bmod eM$, where a_ℓ is the leading coefficient of g
  Append $(a'_\ell \cdot g \cdot (eM)^{m-i})$ to H
end for
$i \leftarrow 1$
for each polynomial $h(x, y, z)$ in H **do**
  Set i-th row of L to the coefficient vector of $h(xX, yY, zZ)$ ordered w.r.t. Ord
  Increment i
end for

Let a_ℓ denote the leading coefficient. Assuming $\gcd(a_\ell, eM) = 1$, we can multiply $g'_{i,j,k}$'s with the inverse a'_ℓ of their corresponding leading coefficient in $\mathbb{Z}/(eM)^m\mathbb{Z}$. Finally, the shift polynomials become

$$h_{i,j,k}(x,y,z) = a'_\ell \cdot g'_{i,j,k}(x,y,z) \cdot (eM)^{m-i}$$

which carry the root (x_0, y_0, z_0) modulo $(eM)^m$.

We let the coefficient vectors of $h_{i,j,k}(xX, yY, zZ)$ represent the basis vectors of a lattice L. Generation of L is summarized in Algorithm 1.

Note that each polynomial in H generated by Algorithm 1 introduces exactly one new monomial, which is appended to Ord that defines the monomial ordering. Hence, the matrix representing the lattice is lower triangular when each row is ordered with respect to Ord. As a result, the determinant of L is the product of the diagonal entries of the representation matrix.

$$\begin{aligned} det(L) = &\left(\prod_{i=0}^{m} \prod_{j=1}^{m-i} \prod_{k=j}^{j+2r-2} X^{i+j} Y^{k+ri-rl_1} Z^{j+t_1-l_1} (eM)^{m-i} \right) \\ &\times \left(\prod_{i=0}^{m} \prod_{k=0}^{t_2} X^{i} Y^{k+ri-rl_2} Z^{t_1-l_2} (eM)^{m-i} \right), \end{aligned}$$

where $l_1 = \min\left\{\lfloor \frac{k+ri}{r} \rfloor, j+a\right\}$ and $l_2 = \min\left\{\lfloor \frac{k+ri}{r} \rfloor, a\right\}$. Letting s_x, s_y, s_z and s_{eM} be the powers of X, Y, Z and eM in $det(L)$, respectively, and denoting the dimension of the lattice by w, we obtain

$$w = \sum_{i=0}^{m} \sum_{j=1}^{m-i} \sum_{k=j}^{j+2r-2} 1 + \sum_{i=0}^{m} \sum_{k=0}^{t_2} 1 = \frac{2r-1}{2} m^2 + t_2 m + o(m^2)$$

$$s_x = \sum_{i=0}^{m} \sum_{j=1}^{m-i} (2r-1)(i+j) + \sum_{i=0}^{m} \sum_{k=0}^{t_2} i = \frac{2r-1}{3} m^3 + \frac{t}{2} m^2 + o(m^3)$$

$$s_{eM} = \sum_{i=0}^{m} \sum_{j=1}^{m-i} (2r-1)(m-i) + \sum_{i=0}^{m} \sum_{k=0}^{t_2} (m-i) = \frac{2r-1}{3} m^3 + \frac{t}{2} m^2 + o(m^3)$$

Assuming $\frac{t_2}{r} \leq t_1 \leq m$, we get as an asymptotic result

$$\begin{aligned} s_y &= \sum_{i=0}^{m} \sum_{j=1}^{m-i} \sum_{k=j}^{j+2r-2} (k+ri-rl_1) + \sum_{i=0}^{m} \sum_{k=0}^{t_2} (k+ri-rl_2) \\ &\approx \frac{1}{2} \left(\frac{r^2 m^3}{3} - r^2 m^2 t_1 + r^2 m t_1^2 - \frac{r^2 t_1^3}{3} + r m^2 t_2 \right. \\ &\qquad \left. -2rmt_1 t_2 + r t_1^2 t_2 + m t_2^2 - t_1 t_2^2 + \frac{t_2^3}{3r} \right) + o(m^3) \end{aligned}$$

$$s_z = \sum_{i=0}^{m} \sum_{j=1}^{m-i} \sum_{k=j}^{j+2r-2} (j + t_1 - l_1) + \sum_{i=0}^{m} \sum_{k=0}^{t_2} (t_1 - l_2)$$
$$\approx \frac{1}{2} \left(\frac{(r-1)^2 m^3}{3r} + (r-1)^2 m^2 t_1 + r m t_1^2 \right.$$
$$\left. - \frac{r t_1^3}{3} + t_1^2 t_2 - \frac{t_1 t_2^2}{r} + \frac{t_2^3}{3r^2} \right) + o(m^3)$$

which are approximated as in [20].
Neglecting the low order terms as similarly done in related works, the conditions in Theorem 2 and Theorem 3 can be simplified to $det(L) < (eM)^{wm}$. In our case, we need

$$s_x(\alpha + \beta - 1) + (s_y + s_z) \left(\frac{1}{r+1} \right) + (s_{eM} - wm)(\alpha + \delta) < 0.$$

to be satisfied. Plugging in the values for s_x, s_y, s_z and s_{eM}, we obtain a polynomial $\rho'(r, \alpha, \beta, \delta)$ with parameters t_1, t_2 and m. Let $t_1 = \tau_1 m$ and $t_2 = \tau_2 m$, and terms of $o(m^3)$ contribute to an error term ϵ. Next, we take the partial derivative of ρ' with respect to τ_1 and τ_2, and find the values making the derivatives zero to obtain the maximum value of ρ'. Finally, for $\gamma := \beta - \delta$, when $\tau_1 = \frac{1 - r\gamma + r^2(1-\gamma)}{2r}$ and

$$\tau_2 = \frac{1 + r^3(1-\gamma) - r^2(1+2\gamma) + r(1-\gamma) + 2r\sqrt{r^2(1-\gamma) + r(1-2\gamma) + 1 - \gamma}}{2r+2}$$

both derivatives become zero. Plugging in these values in ρ', we get a function $\rho(r, \alpha, \beta, \delta)$. When the tuple $(r, \alpha, \beta, \delta)$ satisfy $\rho(r, \alpha, \beta, \delta) < 0$, Howgrave-Graham's theorem is satisfied. We can extract the root (k, p, q) under Assumption 1, and thus factor N in time polynomial in $\log N$. □

Remark 1. We note that our definition of shift polynomials is similar to the one in [20]. The difference is that we work modulo eM instead of modulo e. Hence, the constant coefficient of f_{eM} changes. Equating $M = 1$ (i.e., $\delta = 0$), we obtain the result of Sarkar [20] as a corollary of Theorem 1.

Unfortunately, the exact expression of ρ is too complicated to be stated here. Thus, in Table 1, we provide some numerical values for δ which yields $\rho < 0$ when β is fixed to 1. We remind that for $r = 2$ new attack regions are given in Table 1 when either d or e is full-sized.

4 Experimental Results

In this section, we provide various experimental results. In all of our experiments, we fix d to be full-sized (i.e., $\beta = 1$) which is mostly the case in real-life applications. The values for p, q and d are chosen randomly (or d is the inverse

r	smallest δ value satisfying $\rho(r) < 0$ for $\alpha = 1$	smallest δ value satisfying $\rho(r) < 0$ for $\alpha = 0$
2	0.828	0.362
3	0.798	0.344
4	0.750	0.314
5	0.703	0.285
6	0.662	0.259
7	0.625	0.237

Table 1. Numerical values satisfying $\rho < 0$ for different r and α values where $\beta = 1$.

of $2^{16}+1$ modulo $\phi(N)$). The experiments are performed on Sage 6.5 running on Ubuntu 14.04 LTS with Intel Core i7-3770 CPU at 3.40GHz and 16GB RAM.

Our results are given in Tables 2 and 3. In all of our experiments, Gröbner basis computation yields to a polynomial of the form $y-p$ giving the factorization of N. For the case when $\alpha = \beta = 1$ (which is illustrated in Table 2), we would like to highlight that our result in a case is better than the theoretical bound $\delta \geq 0.828$. However, when e is chosen small (e.g., $e = 2^{16}+1$), the modulus eM becomes very small when compared to the case $\alpha = \beta = 1$. Therefore, the low order terms ignored to simplify the condition to $det(L) < (eM)^{wm}$ have much higher effect in this case. Thus, the results are a little bit worse than the best possible bound of Theorem 1.

r	m	t_1	t_2	w	δ	LLL time (secs)	Gröbner Basis time (secs)
2	6	4	7	119	0.870	1930.21	3.00
2	7	4	8	156	0.860	6517.26	67.99
2	8	4	7	180	0.850	19619.96	1227.18
2	8	5	9	198	0.835	28684.34	358.80
2	9	5	9	235	0.830	63748.97	635.33
2	9	5	10	245	0.823	67480.18	149.56
3	7	4	9	220	0.952	26671.68	7358.66
2	8	5	9	198	0.840	90981.76	2246.77

Table 2. Experimental results for $\alpha = \beta = 1$. $n = 2048$ bits for the last row and $n = 1024$ bits for the rest.

5 Conclusion and Discussion

In this paper, we show a new partial key exposure attack on multi-power RSA, where $N = p^r q$. The attack takes advantage of known LSBs. Our result in Theorem 1 generalizes the work of Sarkar [20]. Moreover, we provide experimental

r	m	t_1	t_2	w	δ	LLL time (secs)	Gröbner Basis time (secs)
2	8	3	2	135	0.520	21234.57	4114.00
2	8	3	2	135	0.510	19082.57	4280.77
2	9	3	3	175	0.500	48950.79	9134.06
2	10	3	2	198	0.485	84090.70	15927.35
3	9	3	3	265	0.510	148030.34	56230.82
2	10	3	2	198	0.500	203293.58	45573.57
2	10	3	2	198	0.490	185964.22	40817.77

Table 3. Experimental results for $e = 2^{16} + 1$, $\beta = 1$. $n = 2048$ bits for the last two rows and $n = 1024$ bits for the rest.

results justifying our claims. Our attack even works in the case when $e, d \approx N$. In fact, our experimental result is better than the theoretical bound for this case. This paves the way for a further study: investigating sublattices of the original lattice to improve the theoretical bound. However, this is a hard task because in this case the lattice will not be of full rank and calculating the determinant gets complicated.

One may wonder why our attack is not directly applicable to known MSBs case. Suppose that we know an MSB part d_0 of d. Then, we obtain the equation

$$ed_0 + e\tilde{d} - 1 = k(p^r - p^{r-1})(q-1),$$

where $\tilde{d}$ represents the unknown part of d. Considering this equation as a polynomial, we get

$$F(w, x, y, z) = 1 - ed_0 - ew + x(N - y^r - y^{r-1}z + y^{r-1}).$$

Now e, N or ed_0 are possible choices of moduli. The case e is studied in [20] where one cannot benefit from partial knowledge of d as it vanishes. If N is chosen as the modulus, then the trick of replacing each term $y^r z$ with N and finding its inverse cannot be applied. That leaves us with the option to choose ed_0 as the modulus. This case actually corresponds to finding a small root of *integer* equations [4], not modular equations [5].

Observe that reducing F modulo ed_0 does not eliminate any variable. In particular, F_{ed_0} and F have the same monomials. Hence, the polynomials derived from LLL may just be those of the form $F \cdot g_i$ for nonzero polynomials g_i not carrying the desired root. More concretely, the attacker does not obtain any additional information at all although LLL-reduced polynomials carry the root since they have the factor F.

For a recent work, one may see Coron's works [7,8] about methods to ensure independence between the initial polynomial F and the polynomials derived after LLL reduction[5]. Unfortunately, the tricks used in this work cannot be directly applied with Coron's method. This issue raises questions about the validity of

[5] This independence is also ensured in Coppersmith's method [4].

known MSBs attack shown in [12]. The authors do not specify any methodology guaranteeing the independence aforementioned. Their experiments for this case are very far away from the new attack region described by Theorem 1 in their paper. Moreover, the authors also state that in some experiments, they just verified that the LLL-reduced polynomials contain the root. As we explained earlier, this does not have any implication for an attacker to be able to find the root.

Acknowledgments: Uzunkol's research is supported by the project (114C027) funded by EU FP7-The Marie Curie Action and TÜBİTAK (2236-CO-FUNDED Brain Circulation Scheme). His work is also partly supported by a joint research project funded by Bundesministerium für Bildung und Forschung (BMBF), Germany (01DL12038) and TÜBİTAK, Turkey (TBAG-112T011).

References

1. Johannes Blömer and Alexander May. New partial key exposure attacks on RSA. In Dan Boneh, editor, *Advances in Cryptology - CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 27–43. Springer Berlin Heidelberg, 2003.
2. D. Boneh and G. Durfee. Cryptanalysis of RSA with private key d less than $N^{0.292}$. *Information Theory, IEEE Transactions on*, 46(4):1339–1349, Jul 2000.
3. Dan Boneh, Glenn Durfee, and Yair Frankel. An attack on RSA given a small fraction of the private key bits. In Kazuo Ohta and Dingyi Pei, editors, *Advances in Cryptology - ASIACRYPT '98*, volume 1514 of *Lecture Notes in Computer Science*, pages 25–34. Springer Berlin Heidelberg, 1998.
4. Don Coppersmith. Finding a small root of a bivariate integer equation; factoring with high bits known. In Ueli Maurer, editor, *Advances in Cryptology - EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 178–189. Springer Berlin Heidelberg, 1996.
5. Don Coppersmith. Finding a small root of a univariate modular equation. In Ueli Maurer, editor, *Advances in Cryptology - EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 155–165. Springer Berlin Heidelberg, 1996.
6. Don Coppersmith. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. *Journal of Cryptology*, 10(4):233–260, 1997.
7. Jean-Sébastien Coron. Finding small roots of bivariate integer polynomial equations revisited. In Christian Cachin and Jan L. Camenisch, editors, *Advances in Cryptology - EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 492–505. Springer Berlin Heidelberg, 2004.
8. Jean-Sébastien Coron. Finding small roots of bivariate integer polynomial equations: A direct approach. In Alfred Menezes, editor, *Advances in Cryptology - CRYPTO 2007*, volume 4622 of *Lecture Notes in Computer Science*, pages 379–394. Springer Berlin Heidelberg, 2007.
9. Matthias Ernst, Ellen Jochemsz, Alexander May, and Benne de Weger. Partial key exposure attacks on RSA up to full size exponents. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 371–386. Springer Berlin Heidelberg, 2005.
10. Jean Charles Faugère. A new efficient algorithm for computing Gröbner Bases without reduction to zero (F5). In *Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*, ISSAC '02, pages 75–83, New York, NY, USA, 2002. ACM.

11. Nicholas Howgrave-Graham. Finding small roots of univariate modular equations revisited. In Michael Darnell, editor, *Crytography and Coding*, volume 1355 of *Lecture Notes in Computer Science*, pages 131–142. Springer Berlin Heidelberg, 1997.
12. Zhangjie Huang, Lei Hu, Jun Xu, Liqiang Peng, and Yonghong Xie. Partial key exposure attacks on Takagi's variant of RSA. In Ioana Boureanu, Philippe Owesarski, and Serge Vaudenay, editors, *Applied Cryptography and Network Security*, volume 8479 of *Lecture Notes in Computer Science*, pages 134–150. Springer International Publishing, 2014.
13. Kouichi Itoh, Noboru Kunihiro, and Kaoru Kurosawa. Small secret key attack on a variant of RSA (due to Takagi). In Tal Malkin, editor, *Topics in Cryptology - CT-RSA 2008*, volume 4964 of *Lecture Notes in Computer Science*, pages 387–406. Springer Berlin Heidelberg, 2008.
14. Marc Joye and Tancrède Lepoint. Partial key exposure on RSA with private exponents larger than N. In Mark D. Ryan, Ben Smyth, and Guilin Wang, editors, *Information Security Practice and Experience*, volume 7232 of *Lecture Notes in Computer Science*, pages 369–380. Springer Berlin Heidelberg, 2012.
15. Paul C. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In Neal Koblitz, editor, *Advances in Cryptology - CRYPTO '96*, volume 1109 of *Lecture Notes in Computer Science*, pages 104–113. Springer Berlin Heidelberg, 1996.
16. A.K. Lenstra, Jr. Lenstra, H.W., and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982.
17. Yao Lu, Rui Zhang, and Dongdai Lin. New results on solving linear equations modulo unknown divisors and its applications. Cryptology ePrint Archive, Report 2014/343, 2014. `http://eprint.iacr.org/`.
18. Alexander May. Secret exponent attacks on RSA-type schemes with moduli $N = p^r q$. In Feng Bao, Robert Deng, and Jianying Zhou, editors, *Public Key Cryptography - PKC 2004*, volume 2947 of *Lecture Notes in Computer Science*, pages 218–230. Springer Berlin Heidelberg, 2004.
19. Phong Q. Nguyen and Damien Stehlé. Floating-Point LLL revisited. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 215–233. Springer Berlin Heidelberg, 2005.
20. Santanu Sarkar. Small secret exponent attack on RSA variant with modulus $N = p^r q$. *Designs, Codes and Cryptography*, 73(2):383–392, 2014.
21. Tsuyoshi Takagi. Fast RSA-type cryptosystem modulo $p^k q$. In Hugo Krawczyk, editor, *Advances in Cryptology - CRYPTO '98*, volume 1462 of *Lecture Notes in Computer Science*, pages 318–326. Springer Berlin Heidelberg, 1998.
22. Michael J. Wiener. Cryptanalysis of short RSA secret exponents. *IEEE Transactions on Information Theory*, 36:553–558, 1990.