How to Factor N_1 and N_2 When $p_1 = p_2 \bmod 2^t$

Kaoru Kurosawa and Takuma Ueda

Ibaraki University, Japan
kurosawa@mx.ibaraki.ac.jp

Abstract. Let $N_1 = p_1 q_1$ and $N_2 = p_2 q_2$ be two different RSA moduli. Suppose that

$$p_1 = p_2 \bmod 2^t$$

for some t, and q_1 and q_2 are α bit primes. Then May and Ritzenhofen showed that N_1 and N_2 can be factored in quadratic time if

$$t \geq 2\alpha + 3.$$

In this paper, we improve this lower bound on t. Namely we prove that N_1 and N_2 can be factored in quadratic time if

$$t \geq 2\alpha + 1.$$

Further our simulation result shows that our bound is tight as far as the factoring method of May and Ritzenhofen is used.

Keywords: factoring, Gaussian reduction algorithm, lattice.

1 Introduction

Factoring $N = pq$ is a fundamental problem in modern cryptography, where p and q are large primes. Since RSA was invented, some factoring algorithms which run in subexponential time have been developed, namely the quadratic sieve [10], the elliptic curve [4] and number field sieve [5]. However, no polynomial time algorithm is known.

On the other hand, the so called oracle complexity of the factorization problem were studied by Rivest and Shamir [11], Maurer [6] and Coppersmith [1]. In particular, Coppersmith [1] showed that one can factor N if a half of the most significant bits of p are given.

Recently, May and Ritzenhofen [7] considered another approach (which received the "Best Paper Award" of PKC 2009). Suppose that we are given $N_1 = p_1 q_1$ and $N_2 = p_2 q_2$. If

$$p_1 = p_2,$$

then it is easy to factor N_1, N_2 by using Euclidean algorithm. May and Ritzenhofen showed that it is easy to factor N_1, N_2 even if

$$p_1 = p_2 \bmod 2^t$$

K. Sakiyama and M. Terada (Eds.): IWSEC 2013, LNCS 8231, pp. 217–225, 2013.

for sufficiently large t. More precisely suppose that q_1 and q_2 are α bit primes. Then they showed that N_1 and N_2 can be factored in quadratic time if

$$t \geq 2\alpha + 3.$$

In this paper, we improve the above lower bound on t. We prove that N_1 and N_2 can be factored in quadratic time if

$$t \geq 2\alpha + 1.$$

Further our simulation result shows that our bound is tight as far as the factoring method of May and Ritzenhofen [7] is used.

Also our proof is conceptually simpler than that of May and Ritzenhofen [7]. In particular, we do not use the Minkowski bound whereas it is required in their proof.

As written in [7], one application of our result is malicious key generation of RSA moduli, i.e. the construction of backdoored RSA moduli [2,13]. In [7], the authors also suggest the following constructive cryptographic applications. Consider the one more RSA modulus problem such that on input $N_1 = p_1 q_1$, one has to produce $N_2 = p_1 q_2$ with $p_1 = p_2 \bmod 2^t$. Our result shows that this problem is equivalent to the factorization problem as long as $t \geq 2\alpha + 1$. So the one more RSA modulus problem might serve as a basis for various cryptographic primitives, whose security is then in turn directly based on factoring (imbalanced) integers.

(Related work) Sarkar and Maitra [12] extended the result of May and Ritzenhofen [7] under a *heuristic* assumption (see Assumption 1 of [12, page 4003]). However, this assumption is *heuristic* only as they wrote in [12].

2 Preliminaries

2.1 Lattice

An integer lattice L is a discrete additive subgroup of Z^n. An alternative equivalent definition of an integer lattice can be given via a basis. Let d, n be integers such that $0 < d \leq n$. Let $\mathbf{b}_1, \cdots, \mathbf{b}_d \in Z^n$ be linearly independent vectors. Then the set of all integer linear combinations of the $\mathbf{b}_i$ spans an integer lattice L, i.e.

$$L = \left\{ \sum_{i=1}^{d} a_i \mathbf{b}_i \mid a_i \in Z \right\}.$$

We call $B = \begin{pmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_d \end{pmatrix}$ a basis of the lattice, the value d denotes the dimension or rank of the basis. The lattice is said to have full rank if $d = n$. The determinant $\det(L)$ of a lattice is the volume of the parallelepiped spanned by the basis

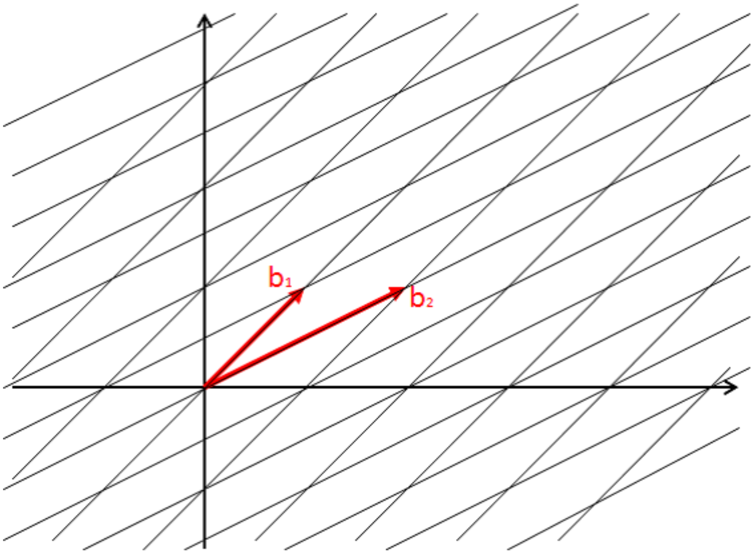

Fig. 1. Lattice

vectors. The determinant $\det(L)$ is invariant under unimodular basis transformations of B. In case of a full rank lattice $\det(L)$ is equal to the absolute value of the Gramian determinant of the basis B. Let us denote by $||\mathbf{v}||$ the Euclidean ℓ_2-norm of a vector $\mathbf{v}$. Hadamardfs inequality [8] relates the length of the basis vectors to the determinant.

Proposition 1. *Let* $B = \begin{pmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_d \end{pmatrix} \in Z^{n\times n}$ *be an arbitrary non-singular matrix. Then*

$$\det(B) \leq \prod_{i=1}^{n} ||\mathbf{b}_i||.$$

The successive minima λ_i of the lattice L are defined as the minimal radius of a ball containing i linearly independent lattice vectors of L (see Fig.2).

Proposition 2. *(Minkowski [9]). Let $L \subseteq Z^n$ be an integer lattice. Then L contains a non-zero vector* $\mathbf{v}$ *with*

$$||\mathbf{v}|| = \lambda_1 \leq \sqrt{n}\det(L)^{1/n}$$

2.2 Gaussian Reduction Algorithm

In a two-dimensional lattice L, basis vectors $\mathbf{v}_1, \mathbf{v}_2$ with lengths $||\mathbf{v}_1|| = \lambda_1$ and$||\mathbf{v}_2|| = \lambda_2$ are efficiently computable by using Gaussian reduction algorithm.

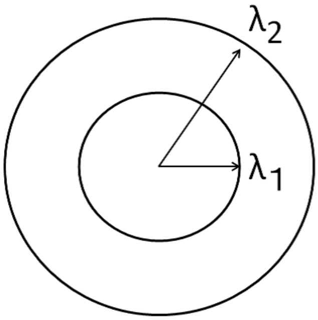

Fig. 2. Successive minima λ_1 and λ_2

Let $\lfloor x \rceil$ denote the nearest integer to x. Then Gaussian reduction algorithm is described as follows.

(Gaussian reduction algorithm)
Input: Basis $\mathbf{b}_1, \mathbf{b}_2 \in Z^2$ for a lattice L.
Output: Basis $(\mathbf{v}_1, \mathbf{v}_2)$ for L such that $||\mathbf{v}_1|| = \lambda_1$ and $||\mathbf{v}_2|| = \lambda_2$.

1. Let $\mathbf{v}_1 := \mathbf{b}_1$ and $\mathbf{v}_2 := \mathbf{b}_2$.
2. Compute $\mu := (\mathbf{v}_1, \mathbf{v}_2)/||\mathbf{v}_1||^2$,
   $\mathbf{v}_2 := \mathbf{v}_2 - \lfloor \mu \rceil \cdot \mathbf{v}_1$.
3. while $||\mathbf{v}_2|| < ||\mathbf{v}_1||$ do:
4. Swap $\mathbf{v}_1$ and $\mathbf{v}_2$.
5. Compute $\mu := (\mathbf{v}_1, \mathbf{v}_2)/||\mathbf{v}_1||^2$,
   $\mathbf{v}_2 := \mathbf{v}_2 - \lfloor \mu \rceil \cdot \mathbf{v}_1$.
6. end while
7. return $(\mathbf{v}_1, \mathbf{v}_2)$.

Proposition 3. *The above algorithm outputs a basis $(\mathbf{v}_1, \mathbf{v}_2)$ for L such that $||\mathbf{v}_1|| = \lambda_1$ and $||\mathbf{v}_2|| = \lambda_2$. Further they can be determined in time $O(\log^2(\max\{||\mathbf{b}_1||, ||\mathbf{b}_2||\})$.*

Information on Gaussian reduction algorithm and its running time can be found in [8,3].

3 Previous Implicit Factoring of Two RSA Moduli

Let $N_1 = p_1 q_1$ and $N_2 = p_2 q_2$ be two different RSA moduli. Suppose that

$$p_1 = p_2 (= p) \bmod 2^t \tag{1}$$

for some t, and q_1 and q_2 are α bit primes. This means that p_1, p_2 coincide on the t least significant bits. I.e.,

$$p_1 = p + 2^t \tilde{p}_1 \text{ and } p_2 = p + 2^t \tilde{p}_2$$

for some common p that is unknown to us. Then May and Ritzenhofen [7] showed that N_1 and N_2 can be factored in quadratic time if $t \geq 2\alpha + 3$. In this section, we present their idea.

From eq.(1), we have

$$N_1 = pq_1 \bmod 2^t$$
$$N_2 = pq_2 \bmod 2^t$$

Since q_1, q_2 are odd, we can solve both equations for p. This leaves us with

$$N_1/q_1 = N_2/q_2 \bmod 2^t$$

which we write in form of the linear equation

$$(N_2/N_1)q_1 - q_2 = 0 \bmod 2^t \quad (2)$$

The set of solutions

$$L = \{(x_1, x_2) \in Z^2 \mid (N_2/N_1)x_1 - x_2 = 0 \bmod 2^t\}$$

forms an additive, discrete subgroup of Z^2. Thus, L is a 2-dimensional integer lattice. L is spanned by the row vectors of the basis matrix

$$B_L = \begin{pmatrix} 1, & (N_2/N_1 \bmod 2^t) \\ 0, & 2^t \end{pmatrix} \quad (3)$$

The integer span of B_L, denoted by $span(B_L)$, is equal to L. To see why, let

$$\mathbf{b}_1 = (1, (N_2/N_1 \bmod 2^t))$$
$$\mathbf{b}_2 = (0, 2^t)$$

Then they are solutions of

$$(N_2/N_1)x_1 - x_2 = 0 \bmod 2^t$$

Thus, every integer linear combination of $\mathbf{b}_1$ and $\mathbf{b}_2$ is a solution which implies that $span(B_L) \subseteq L$.

Conversely, let $(x_1, x_2) \in L$, i.e.

$$(N_2/N_1)x_1 - x_2 = k \cdot 2^t$$

for some $k \in Z$. Then

$$(x_1, -k)B_L = (x_1, x_2) \in span(B_L)$$

and thus $L \subseteq span(B_L)$.

Notice that by Eq. (2), we have

$$\mathbf{q} = (q_1, q_2) \in L. \quad (4)$$

If we were able to find this vector in L, then we could factor N_1, N_2 easily. We know that the length of the shortest vector is upper bounded by the Minkowski bound

$$\sqrt{2} \cdot \det(L)^{1/2} = \sqrt{2} \cdot 2^{t/2}.$$

Since we assume that q_1, q_2 are α-bit primes, we have $q_1, q_2 \le 2^\alpha$. If α is sufficiently small, then $||\mathbf{q}||$ is smaller than the Minkowski bound and, therefore, we can expect that q is among the shortest vectors in L. This happens if

$$||\mathbf{q}|| \le \sqrt{2} \cdot 2^\alpha \le \sqrt{2} \cdot 2^{t/2}$$

So if $t \ge 2\alpha$, we expect that $\mathbf{q}$ is a short vector in L. We can find a shortest vector in L using Gaussian reduction algorithm on the lattice basis B in time

$$O(\log^2(2^t)) = O(\log^2(\min\{N_1, N_2\})).$$

By elaborating the above argument, May and Ritzenhofen [7] proved the following.

Proposition 4. *Let $N_1 = p_1 q_1$ and $N_2 = p_2 q_2$ be two different RSA moduli such that $p_1 = p_2 \bmod 2^t$ for some t, and q_1 and q_2 are α bit primes. If*

$$t \ge 2\alpha + 3, \tag{5}$$

then N_1, N_2 can be factored in time $O(\log^2(\min\{N_1, N_2\}))$.

4 Improvement

In this section, we improve the lower bound on t of Proposition 4.

Lemma 1. *If $||\mathbf{q}|| < \lambda_2$, then $\mathbf{q} = c \cdot \mathbf{v}_1$ for some integer c, where $\mathbf{v}_1$ is the shortest vector in L.*

(Proof) Suppose that $\mathbf{q} \ne c \cdot \mathbf{v}_1$ for any integer c. This means that $\mathbf{v}_1$ and $\mathbf{q}$ are linearly independent vectors. Therefore it must be that $||\mathbf{q}|| \ge \lambda_2$ from the definition of λ_2. However, this is against our assumption that $||\mathbf{q}|| < \lambda_2$. Therefore we have $\mathbf{q} = c \cdot \mathbf{v}_1$ for some integer c.

Q.E.D.

Lemma 2. *If q_1 and q_2 are α bits long, then*

$$||\mathbf{q}|| < 2^{\alpha+0.5}$$

(Proof) Since q_1 and q_2 are α-bits long, we have

$$q_i \le 2^\alpha - 1$$

for $i = 1, 2$. Therefore

$$||\mathbf{q}|| \le \sqrt{2}(2^\alpha - 1) < \sqrt{2} \cdot 2^\alpha = 2^{\alpha+0.5}$$

Q.E.D.

Theorem 1. *Let $N_1 = p_1q_1$ and $N_2 = p_2q_2$ be two different RSA moduli such that*

$$p_1 = p_2 \bmod 2^t$$

for some t, and q_1 and q_2 are α-bit primes. If

$$t \geq 2\alpha + 1, \tag{6}$$

then N_1, N_2 can be factored in time $O(\log^2(\min\{N_1, N_2\}))$.

(Proof) If $q_1 = q_2$, the we can factor N_1, N_2 by using Euclidean algorithm easily. Therefore we assume that $q_1 \neq q_2$.

Apply Gaussian reduction algorithm to B_L. Then we obtain

$$B_0 = \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{pmatrix}$$

such that

$$||\mathbf{v}_1|| = \lambda_1 \text{ and } ||\mathbf{v}_2|| = \lambda_2.$$

We will show that $\mathbf{q} = \mathbf{v}_1$ or $\mathbf{q} = -\mathbf{v}_1$, where $\mathbf{q} = (q_1, q_2)$.

From Hadamard's inequality, we have

$$||\mathbf{v}_2||^2 \geq ||\mathbf{v}_1||||\mathbf{v}_2|| \geq \det(B_0) = \det(B_L) = 2^t,$$

where $\det(B_0) = \det(B_L)$ because B_0 and B_L span the same lattice L. The last equality comes from eq.(3). Therefore we obtain that

$$\lambda_2 = ||\mathbf{v}_2|| \geq 2^{t/2}.$$

Now suppose that

$$t \geq 2\alpha + 1$$

Then

$$t/2 \geq \alpha + 0.5.$$

Therefore

$$\lambda_2 = ||\mathbf{v}_2|| \geq 2^{t/2} \geq 2^{\alpha+0.5} > ||\mathbf{q}||$$

from Lemma 2. This means that

$$(q_1, q_2) = \mathbf{q} = c \cdot \mathbf{v}_1$$

for some integer c from Lemma 1. Further since $\gcd(q_1, q_2) = 1$, it must be that $c = 1$ or -1. Therefore $\mathbf{q} = \mathbf{v}_1$ or $\mathbf{q} = -\mathbf{v}_1$ (see Fig.3).

Finally from Proposition 3, Gaussian reduction algorithm runs in time

$$O(\log^2(2^t)) = O(\log^2(\min\{N_1, N_2\})).$$

Q.E.D.

Compare eq.(6) and eq.(5), and notice that we have improved the previous lower bound on t.

Also our proof is conceptually simpler than that of May and Ritzenhofen [7]. In particular, we do not use the Minkowski bound whereas it is required in their proof.

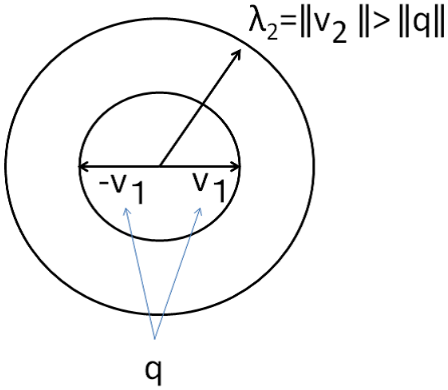

Fig. 3. Proof of Theorem 1

5 Generalization

Theorem 1 can be generalized as follows.

Corollary 1. *Let* $N_1 = p_1 q_1$ *and* $N_2 = p_2 q_2$ *be two different RSA moduli such that*

$$p_1 = p_2 \bmod T$$

for some T*. Let* q_1 *and* q_2 *be* α*-bits long primes. Then if*

$$T \geq 2^{2\alpha+1} \quad (7)$$

then N_1, N_2 *can be factored in time* $O(\log^2(\min\{N_1, N_2\}))$*.*

Corollary 2. *Let* $N_1 = p_1 q_1$ *and* $N_2 = p_2 q_2$ *be two different RSA moduli such that*

$$p_1 = p_2 \bmod T$$

for some T*. If*

$$T > q_1^2 + q_2^2 \quad (8)$$

then N_1, N_2 *can be factored in time* $O(\log^2(\min\{N_1, N_2\}))$*.*

The proofs are almost the same as that of Theorem 1.

6 Simulation

We verified Theorem 1 by computer simulation. We considered the case such that q_1 and q_2 are $\alpha = 250$ bits long. Theorem 1 states that if

$$t \geq 2\alpha + 1 = 501,$$

then we can factor N_1 and N_2 by using Gaussian reduction algorithm. The simulation results are shown in Table 6, where p_1 and p_2 are 750 bits long. For each value of t, the success rate is computed over 100 samples.

From this table, we can see that we can indeed factor N_1 and N_2 if $t \geq 501$. We can also see that we fail to factor N_1 and N_2 if $t \leq 500$. This shows that our bound is tight as far as the factoring method of May and Ritzenhofen [7] is used.

Table 1. Computer Simulation

number of shared bits t	success rate
503	100%
502	100%
501	100%
500	40%
499	0%
498	0%

References

1. Coppersmith, D.: Finding a small root of a bivariate integer equation; factoring with high bits known. In: Maurer, U.M. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 178–189. Springer, Heidelberg (1996)
2. Crépeau, C., Slakmon, A.: Simple backdoors for RSA key generation. In: Joye, M. (ed.) CT-RSA 2003. LNCS, vol. 2612, pp. 403–416. Springer, Heidelberg (2003)
3. Galbraith, S.D.: Mathematics of Public Key Cryptography. Cambridge University Press (2012)
4. Lenstra Jr., H.W.: Factoring Integers with Elliptic Curves. Ann. Math. 126, 649–673 (1987)
5. Lenstra, A.K., Lenstra Jr., H.W.: The Development of the Number Field Sieve. Springer, Heidelberg (1993)
6. Maurer, U.M.: Factoring with an oracle. In: Rueppel, R.A. (ed.) EUROCRYPT 1992. LNCS, vol. 658, pp. 429–436. Springer, Heidelberg (1993)
7. May, A., Ritzenhofen, M.: Implicit factoring: On polynomial time factoring given only an implicit hint. In: Jarecki, S., Tsudik, G. (eds.) PKC 2009. LNCS, vol. 5443, pp. 1–14. Springer, Heidelberg (2009)
8. Meyer, C.D.: Matrix Analysis and Applied Linear Algebra. Cambridge University Press, Cambridge (2000)
9. Minkowski, H.: Geometrie der Zahlen. Teubner-Verlag (1896)
10. Pomerance, C.: The quadratic sieve factoring algorithm. In: Beth, T., Cot, N., Ingemarsson, I. (eds.) EUROCRYPT 1984. LNCS, vol. 209, pp. 169–182. Springer, Heidelberg (1985)
11. Rivest, R.L., Shamir, A.: Efficient factoring based on partial information. In: Pichler, F. (ed.) EUROCRYPT 1985. LNCS, vol. 219, pp. 31–34. Springer, Heidelberg (1986)
12. Sarkar, S., Maitra, S.: Approximate Integer Common Divisor Problem Relates to Implicit Factorization. IEEE Transactions on Information Theory 57(6), 4002–4013 (2011)
13. Young, A., Yung, M.: The prevalence of kleptographic attacks on discrete-log based cryptosystems. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 264–276. Springer, Heidelberg (1997)