Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 650537, 9 pages
http://dx.doi.org/10.1155/2014/650537

Research Article

On the Improvement of Wiener Attack on RSA with Small Private Exponent

Mu-En Wu,[1] Chien-Ming Chen,[2,3] Yue-Hsun Lin,[4] and Hung-Min Sun[5]

[1] *Department of Mathematics, Soochow University, Taipei, Taiwan*
[2] *School of Computer Science and Technology, Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen, China*
[3] *Shenzhen Key Laboratory of Internet Information Collaboration, Shenzhen, China*
[4] *CyLab, Carnegie Mellon University, Pittsburgh, PA 15213, USA*
[5] *Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan*

Correspondence should be addressed to Hung-Min Sun; hmsun@cs.nthu.edu.tw

Received 7 February 2014; Accepted 27 February 2014; Published 27 March 2014

Academic Editors: T. Cao and F. Yu

RSA system is based on the hardness of the integer factorization problem (IFP). Given an RSA modulus $N = pq$, it is difficult to determine the prime factors p and q efficiently. One of the most famous short exponent attacks on RSA is the Wiener attack. In 1997, Verheul and van Tilborg use an exhaustive search to extend the boundary of the Wiener attack. Their result shows that the cost of exhaustive search is $2r + 8$ bits when extending the Weiner's boundary r bits. In this paper, we first reduce the cost of exhaustive search from $2r + 8$ bits to $2r + 2$ bits. Then, we propose a method named EPF. With EPF, the cost of exhaustive search is further reduced to $2r - 6$ bits when we extend Weiner's boundary r bits. It means that our result is 2^{14} times faster than Verheul and van Tilborg's result. Besides, the security boundary is extended 7 bits.

1. Introduction

During the past 30 years, RSA [1] has been one of the most popular public-key cryptosystems worldwide. It has been widely used in several applications [2–4]. The security of RSA is often based on the hardness of the integer factorization problem (IFP), which remains a well-studied problem [5, 6]. Current RSA standards suggest that an RSA modulus N should be at least 1024 bits long. Using the best-known factoring algorithms, the expected workload of factoring a 1024 bit modulus is 2^{80}, which is currently believed to be infeasible. However, although the use of a large RSA modulus achieves a high security level, the encryption and decryption procedures involve heavy exponential modular multiplications, which make RSA inefficient. Therefore, many approaches have been investigated for speeding-up the RSA encryption (or signature-verification) and RSA decryption (or signature-signing) [7–12]. Furthermore, since the signing task is often executed by lightweight devices, such as smart cards, mobile phones, or PDAs, the research on speeding-up signature-signing is more practical and important.

The most popular method for reducing the signing time is to apply a small private exponent d since the complexity of signing depends on the bit-length of d. In order to achieve this goal, the order of choosing e and d is exchanged. d is first chosen in the RSA-key generation algorithm, and the corresponding public exponent e satisfying $ed \equiv 1 \pmod{\varphi(N)}$ is then calculated. These RSA variants are called RSA-Small-d. Nevertheless, the variants of RSA-Small-d have the security flaws [13–18]. In fact, instances of RSA with $d < N^{1/4}$ can be efficiently broken by Wiener attack [16]. Besides, Boneh and Durfee's lattice-based attack [19] indicated that an instance of RSA-Small-d with $d < N^{0.292}$ should be considered to be an unsafe system.

In 1997, Verheul and van Tilborg [20] used an exhaustive search to further extend the boundary of the Wiener attack.

Suppose $r = \log_2 d - \log_2 N^{1/4}$; their result shows that an exhaustive search for $2r + 8$ bits is required to extend the Wiener's boundary r bits. Assume that an exhaustive search for 64 bits is feasible in terms of current computational abilities; solving r for the equation "$2r+8 = 64$" yields $r = 28$, which implies that the boundary of the Wiener attack should be raised up to $N^{1/4}2^{28}$.

In this paper, we attempt to reduce the cost of exhaustive search of Verheul and van Tilborg's result. We propose an approach to reduce the cost of exhaustive search when we desire to extend Wiener's boundary. This approach includes two steps.

Step 1. We investigate a method for searching as many MSBs (most significant bits) of $p+q$ as possible, which is equivalent to estimating $p + q$ as accurately as possible. In this step, to extend Wiener's boundary r bits, an exhaustive search requires $2r + 2$ bits. It means that our result is better than Verheul and van Tilborg's cost, which requires an exhaustive search for $2r + 8$ bits.

Step 2. We develop an approach, called "Estimated Prime Factor (EPF)," to estimate $p + q$, and then we derive two integers p_E and q_E, which are the estimations of p and q, respectively. Using EPF, the first 8 MSBs of $p + q$ can be determined. This result is more accurate than the traditional estimation, which estimates $p+q$ by $2\sqrt{N}$. Applying EPF can further reduce the cost of exhaustive search. More specifically, to extend Wiener's boundary r bits, an exhaustive search requires $2r-6$ bits. As compared to Verheul and van Tilborg's result, which requires an exhaustive search for $2r + 8$ bits, the security boundary is extended 7 bits.

1.1. Our Contribution. The contributions of this paper are summarized as follows.

(1) We first reduce the cost of exhaustive search from $2r+8$ (Verheul and van Tilborg's result) bits to $2r + 2$ bits when we extend Wiener's boundary r bits. It means that exhaustive search is 2^6 times faster in Step 1. Besides, the security boundary is extended 3 bits.

(2) We propose a novel approach, named EPF, for estimating the prime factors of N. With EPF, the cost of the exhaustive search for $2r + 2$ bits (mentioned in contribution (1)) is further reduced to $2r - 6$ bits. Compared with Verheul and van Tilborg's result, exhaustive search is 2^{14} times faster. Besides, the security boundary is extended 7 bits.

1.2. Organization. The remainder of this paper is organized as follows. Section 2 presents the preliminaries of this paper. Section 3 describes Step 1 of our approach. In Section 4, we propose the EPF to estimate the prime factors of an RSA modulus. Next, Step 2 of our approach which is applying EPF is proposed in Section 5. Finally, we present our conclusions and future works in Section 6.

2. Preliminary

In this section, we introduce the preliminaries of this paper which include RSA and its variants and the Wiener attack.

2.1. RSA and Its Variants. The RSA cryptosystem [1] consists of three parts, RSA-key generation, encryption, and decryption which are described as follows.

2.1.1. RSA-Key Generation, Encryption, and Decryption. The RSA-key generation outputs the RSA key: (N, e, d). First, randomly choose two large prime numbers p and q and compute $N = pq$, where N is called RSA modulus. Secondly, let e, called public exponent, be a randomly chosen integer such that $\gcd(e, \varphi(N)) = 1$, where $\varphi(\cdot)$ is Euler's phi function. Then, let d, called private exponent, be the multiplicative inverse modulo $\varphi(N)$ (i.e., $ed \equiv 1 \pmod{\varphi(N)}$). The pair (e, N) is the public key and the pair (d, N) is the private key.

From the key relation $ed \equiv 1 \pmod{\varphi(N)}$, there exists a unique positive integer k satisfying

$$ed = 1 + k \cdot \varphi(N). \tag{1}$$

We call (1) as the RSA-key equation. To encrypt a plaintext message $M \in \mathbb{Z}_N$, compute $C \equiv M^e \pmod{N}$. The result C is called the ciphertext of M. To execute RSA decryption, a ciphertext $C \in \mathbb{Z}_N$ is decrypted by raising it to the dth power modulo N. From Lagrange's theorem, it follows that

$$C^d \left(\mathrm{mod} N\right) = M^{ed} \left(\mathrm{mod} N\right) \equiv M \pmod{N} = M. \tag{2}$$

Usually, one often selects e as small as possible due to the reason of efficient encryption (or signature-verification). The smallest e is suggested to be $2^{32} + 1$ rather than $2^{16} + 1$ while a known affine relation between two messages exists [21]. We call the RSA system with small public exponent e as "RSA-Small-e." On the other hand, since the cost of decryption (or signature-signing) can be significantly reduced when the private exponent d is much smaller than $\varphi(N)$, in order to simply reduce the decryption (or signature-signing) time, one can select a small private exponent d first in RSA-key generation. Such variant is called RSA-Small-d, which is shown in the following.

2.1.2. RSA-Small-d. Generating instances of RSA with a small private exponent is easy with the observation that the RSA-key equation (1) is symmetric with respect to the public and private exponents. We simply follow the same key generation of original RSA but exchange the choosing order of public and private exponents.

One of the drawbacks of RSA-Small-d is its inefficient encryption. Since the public exponent e in RSA-Small-d is always computed as the inverse of d modulo $\varphi(N)$, it is expected with high probability that e will be almost the same size as $\varphi(N)$. In conclusion, RSA-Small-d saves the decryption (or signature) cost while the encryption cost remains large.

2.2. The Wiener Attack. One of the most famous short exponent attacks on RSA is the Wiener attack. Boneh and Durfee [22] showed in 1990 that RSA-Small-d should be considered insecure when $d < N^{1/4}$. He achieved the attack through the technique of continued fractions. In the following paragraph, we briefly introduce the continued fractions and the Weiner attack. The details can be referenced in [16].

Definition 1 (continued fractions). For any positive real number α, define $\alpha = \xi_0$, $a_i = \lfloor \xi_i \rfloor$, $\xi_{i+1} = 1/(\xi_i - a_i)$ for $i = 0, 1, 2, \ldots$. Then α can be expanded into the following form:

$$\alpha_i = a_0 + 1/\left(a_1 + 1/\left(a_2 + 1/\left(a_3 + 1/\cdots\right)\right)\right). \tag{3}$$

The form of (3) is called the continued fraction expression of α. For simplicity, we write (3) to be $\alpha = (a_0, a_1, a_2, \ldots)$. In addition, denote $\alpha_i = (a_0, a_1, \ldots, a_i)$ as the ith convergent of the continued fraction expansion of α, which means

$$\alpha_i = a_0 + 1/\left(a_1 + 1/\left(a_2 + 1/\left(\cdots + 1/a_i\right)\right)\right). \tag{4}$$

If α is a rational number, then the process of computing its continued fraction expression, see (3), will cease in some index k. That is, $\alpha = \alpha_k$. If α is irrational, then the process will go on unceasingly.

Theorem 2. *Denote h_i/k_i as the fraction form of* (4)*; that is, $h_i/k_i = \alpha_i$, where h_i and k_i are positive integers. Then, h_i and k_i can be calculated by defining $h_{-2} = 0$, $k_{-2} = 1$, $h_{-1} = 1$, and $k_{-1} = 0$. And $h_i = a_i h_{i-1} + h_{i-2}$ and $k_i = a_i k_{i-1} + k_{i-2}$, for $i \geq 0$.*

Following the notations in Theorem 2, we have Corollary 3.

Corollary 3. *For any $i \geq 1$,*

$$\left|\alpha - \frac{h_{i+1}}{k_{i+1}}\right| < \left|\alpha - \frac{h_i}{k_i}\right|. \tag{5}$$

Furthermore, if α is an irrational number, then $\lim_{i \to \infty} h_i/k_i = \alpha$.

Theorem 4. *If a real number α and a reduced fraction a/b satisfy*

$$\left|\alpha - \frac{a}{b}\right| < \frac{1}{2b^2}, \tag{6}$$

then a/b equals to one of the convergents of the continued fraction expression of α.

2.2.1. The Wiener Attack. The Wiener attack [16] is based on approximations using continued fractions to find the private exponent of RSA-Small-d in polynomial time if $d < N^{1/4}$, where p and q are of the same bit-length. Note that the RSA-key equation, $ed = 1 + k \cdot \varphi(N)$, can be rewritten as

$$\left|\frac{e}{\varphi(N)} - \frac{k}{d}\right| = \left|\frac{1}{d\varphi(N)}\right|, \tag{7}$$

which is similar to the form of the left-hand side of (6). In order to apply Theorem 4, we replace $e/\varphi(N)$ of (7) by e/N, which is known for everyone, and set the difference between e/N and k/d to be smaller than $1/2d^2$; that is,

$$\left|\frac{e}{N} - \frac{k}{d}\right| < \frac{1}{2d^2}. \tag{8}$$

Therefore, according to Theorem 4, k/d can be found by computing one of the convergents of the continued fraction expression of e/N.

The security boundary of the Wiener attack is deduced from the sufficient condition for (8). Since $p \approx q \approx \sqrt{N}$ and $k \approx d$, the left-hand side of (8) is simplified to

$$\left|\frac{e}{N} - \frac{k}{d}\right| = \frac{k(p+q-1)-1}{Nd} \approx \frac{2\sqrt{N}}{N} = \frac{2}{\sqrt{N}}. \tag{9}$$

Hence, (8) is transformed to

$$\frac{2}{\sqrt{N}} < \frac{1}{2d^2}, \tag{10}$$

which gives the security boundary of the Wiener attack (after ignoring the constant term):

$$d < N^{1/4}. \tag{11}$$

2.3. Verheul and van Tilborg's Extension. The Wiener attack works very well and efficiently when the private exponent $d < N^{1/4}$. However, what about if the bit-length of d is slightly larger than the bit-length of $N^{1/4}$? In 1997, Verheul and van Tilborg [20] proposed a technique to solve this problem by performing an exhaustive search for $2r + 8$ bits, where $r = \log_2 d - \log_2 N^{1/4}$ means that the bit-length of d is longer than the bit-length of $N^{1/4}$ by r bits.

Verheul and van Tilborg observed that k/d in (8) can be represented as follows:

$$\frac{k}{d} = \frac{p_{j+1}U + (U\Delta + V)p_j}{q_{j+1}U + (U\Delta + V)q_j}, \tag{12}$$

where p_i/q_i is the ith convergent of the continued fraction expression of e/N, $\Delta = 1$ or 2, and U and V are two unknown integers with upper bounds as follows:

$$\log_2 U \leq r + 4, \qquad \log_2 V \leq r + 4. \tag{13}$$

Since Δ is a small integer, we can omit its uncertainty. The unknown parts of (12) are about $2r + 8$ bits, which give the result of Verheul and van Tilborg's extension: extending Wiener's boundary by r bits requires an exhaustive search for about $2r + 8$ bits.

Assume that an exhaustive search for 64 bits is feasible in terms of the current computational capabilities. Solving r for the equation "$2r + 8 = 64$" yields $r = 28$, which implies that Wiener's boundary can be extended 28 bits over the bit-length of $N^{1/4}$. Therefore, RSA-Small-d with $d < N^{1/4}2^{28}$ can be totally broken by continued fraction attack plus the cost of

performing an exhaustive search for 64 bits. In Section 3, we show that, in order to extend Wiener's boundary by r bits, it requires only an exhaustive search for $2r + 2$ bits, rather than that from Verheul and van Tilborg's extension for cost, which requires an exhaustive search for $2r + 8$ bits.

3. Reducing the Cost of Exhaustive Search to $2r+2$ Bits

Our approach contains two steps which are described in Sections 3 and 5, respectively. In this section, we investigate a method for searching as many MSBs (most significant bits) of $p+q$ as possible, which is equivalent to estimating $p+q$ as accurately as possible. With this method, we can reduce the cost of exhaustive search from $2r + 8$ bits (Verheul and van Tilborg's extension) to $2r + 2$ bits when we extend Wiener's boundary r bits.

Let A be the estimation of $p + q$. Throughout this paper, we assume $A < p + q$. Thus $\varphi(N) = (N + 1) - (p + q)$ is estimated as $(N + 1) - A$, which implies

$$\frac{e}{\varphi(N)} \approx \frac{e}{(N+1)-A}. \tag{14}$$

Applying (14) to the Wiener attack, that is, replacing e/N of (8) by $e/((N + 1) - A)$, we have

$$\left|\frac{e}{N+1-A} - \frac{k}{d}\right| < \frac{1}{2d^2}. \tag{15}$$

Note that if $A = p+q$, then (15) always holds for any d because

$$\begin{aligned}\left|\frac{e}{N+1-(p+q)} - \frac{k}{d}\right| &= \left|\frac{ed - k(N+1-(p+q))}{(N+1-(p+q))\,d}\right| \\ &= \frac{1}{\varphi(N)\,d} < \frac{1}{2d^2}.\end{aligned} \tag{16}$$

Simplifying (15) yields

$$\begin{aligned}\left|\frac{e}{N+1-A} - \frac{k}{d}\right| &= \left|\frac{ed - k(N+1-A)}{(N+1-A)\,d}\right| \\ &= \frac{k[(p+q)-A]-1}{(N+1-A)\,d} < \frac{1}{2d^2},\end{aligned} \tag{17}$$

which is

$$2dk[(p+q)-A] - 2d < N + 1 - A. \tag{18}$$

Solving d in (18), we get the upper bound of the private exponent:

$$d < \frac{N+1-A}{2k(p+q-A)-2}. \tag{19}$$

According to the above inequality, we know that the smaller the difference between $p + q$ and A, the higher the upper bound of d. Consequently, in order to extend the security boundary of RSA-Small-d, we attempt to estimate A as precisely as possible such that $p + q - A$ becomes small. Equation (19) also shows that the complexity of further extending Wiener's boundary can be reduced to the complexity of estimating the MSBs of $p + q$. The relation is shown in the following.

Rearranging (18) we have

$$2dk(p+q-1) - 2d < N + (2dk-1)(A-1). \tag{20}$$

Denote Λ as the difference between $p + q$ and A. That is, $\Lambda = p + q - A$. Replacing A in (20) by $p + q - \Lambda$ conducts

$$\begin{aligned}2dk(p+q-1) - 2d &< N + (2dk-1)((p+q-\Lambda)-1) \\ &= 2dk(p+q-1) \\ &\quad + \varphi(N) - \Lambda(2dk-1).\end{aligned} \tag{21}$$

In (21), eliminating $2dk(p + q - 1)$ in both sides we get

$$\Lambda(2dk-1) - 2d < \varphi(N). \tag{22}$$

Now we consider the bit-length of each side. Assume that the bit-length of d is $n/4 + r$ bits, which is longer than Wiener's boundary by r bits. Due to the key generation of RSA-Small-d, the parameter k is almost the same size as d with a high probability; that is, $\log_2 k \approx \log_2 d$. In addition, we perform an exhaustive search for the first s MSBs of $p + q$. Thus the difference between $p+q$ and A can be reduced to $(n/2+1)-s$ bits; that is, $\log_2 \Lambda \approx (n/2 + 1) - s$. Consequently, The term $\Lambda \cdot 2dk$, which dominates the size in the left-hand side of (22), is about $((n/2 + 1) - s) + 1 + 2 \times (n/4 + r)$ bits long and the sufficient condition of (22) is

$$\underbrace{((n/2+1)-s)}_{\text{for } \Lambda} + \underbrace{1 + 2 \times (n/4+r)}_{\text{for } 2dk} < n, \tag{23}$$

which is simplified to

$$2r + 2 < s. \tag{24}$$

Equation (24) gives the following conclusion. In order to extend Wiener's boundary by r bits, we have to perform an exhaustive search for the first $2r+2$ MSBs of $p+q$, where $r = \log_2 d - \log_2 N^{1/4}$. This result is better than that of Verheul and van Tilborg's cost [20], which requires an exhaustive search for $2r+8$ bits. Therefore, assume that an exhaustive search for 64 bits is feasible in terms of current computational abilities. Solving r for

$$2r + 2 = 64 \tag{25}$$

yields $r = 31$, which means that RSA-Small-d is insecure when $d < N^{1/4}2^{31}$.

4. Estimated Prime Factor (EPF)

In this section, a novel approach called Estimated Prime Factor (EPF), which is used to estimate the prime factors of an RSA modulus N, is proposed.

4.1. EPF. Without loss of generality, we assume that $q < p < 2q$, where $N = pq$. Denote D_p and D_q as the distances between $\sqrt{N}$ & p and q & $\sqrt{N}$, respectively. That is,

$$p = \sqrt{N} + D_p, \qquad q = \sqrt{N} - D_q. \tag{26}$$

Applying (26) to $N = pq$ yields

$$N = p \cdot q = \left(\sqrt{N} + D_p\right) \cdot \left(\sqrt{N} - D_q\right) \tag{27}$$

$$= N + \sqrt{N} \cdot \left(D_p - D_q\right) - D_p \cdot D_q. \tag{28}$$

Eliminating N in both sides of (27) we have

$$D_p \cdot D_q = \sqrt{N} \cdot \left(D_p - D_q\right), \tag{29}$$

which leads to

$$\frac{1}{\sqrt{N}} = \frac{D_p - D_q}{D_p D_q}. \tag{30}$$

Equation (30) is quite interesting because the irrational fraction $1/\sqrt{N}$ reveals partial information of $D_p - D_q$ and $D_p \cdot D_q$. Note that with $D_p - D_q$ and $D_p \cdot D_q$ we can compute $D_p + D_q$ by

$$\left(D_p + D_q\right)^2 = \left(D_p - D_q\right)^2 + 4D_p D_q \tag{31}$$

and solve D_p and D_q as follows:

$$\begin{aligned} D_p &= \frac{D_p + D_q}{2} + \frac{D_p - D_q}{2}, \\ D_q &= \frac{D_p + D_q}{2} - \frac{D_p - D_q}{2}. \end{aligned} \tag{32}$$

Now we use continued fractions to construct a rational sequence to approximate $1/\sqrt{N}$. Suppose that the ith convergent of the continued fraction expansion of $1/\sqrt{N}$ is h_i/k_i. According to Theorem 2, we know that

$$\frac{h_i}{k_i} \longrightarrow \frac{1}{\sqrt{N}}, \quad \text{as } i \longrightarrow \infty. \tag{33}$$

Since the sizes of h_i and k_i grow with increase of the index i (see Theorem 2), there exists an index t such that

$$h_t < D_p - D_q < h_{t+1}. \tag{34}$$

We use h_t and k_t as the estimations of $D_p - D_q$ and $D_p D_q$, respectively, instead of using the larger ones. That is,

$$h_t \approx D_p - D_q, \qquad k_t \approx D_p D_q. \tag{35}$$

From (31), $D_p + D_q$ is estimated as

$$D_p + D_q \approx \sqrt{h_t^2 + 4k_t}. \tag{36}$$

And thus D_p and D_q are estimated as

$$D_p \approx \frac{\sqrt{h_t^2 + 4k_t} + h_t}{2}, \qquad D_q \approx \frac{\sqrt{h_t^2 + 4k_t} - h_t}{2}. \tag{37}$$

Finally, we define the estimated prime factors of N as

$$\begin{aligned} p_E &:= \left\lfloor \sqrt{N} + \frac{\sqrt{h_t^2 + 4k_t} + h_t}{2} \right\rfloor, \\ q_E &:= \left\lfloor \sqrt{N} - \frac{\sqrt{h_t^2 + 4k_t} - h_t}{2} \right\rfloor. \end{aligned} \tag{38}$$

4.2. Theoretical Estimation and Experimental Result on Searching the Index t. The process of computing the convergent of the continued fraction expression of $1/\sqrt{N}$ should be ceased at the index t satisfying (34). Thus, we have to estimate the size of $D_p - D_q$ in order to determine the index t. Since $D_p < p$ and $D_q < q$, h_t should not be set larger than $n/2$ bits at least. Next, we investigate the method to estimate the index t theoretically and experimentally.

4.2.1. Theoretical Estimation. From the definitions of D_p and D_q in (26), we have

$$D_p - D_q = p + q - 2\sqrt{N} = \left(\sqrt{p} - \sqrt{q}\right)^2, \tag{39}$$

which is equivalent to

$$\log_2\left(D_p - D_q\right) = 2\log_2\left(\sqrt{p} - \sqrt{q}\right). \tag{40}$$

Equation (40) shows that the bit-length of $D_p - D_q$ is twice the bit-length of $\sqrt{p} - \sqrt{q}$. Consider the following problem.

Problem. Randomly select two prime numbers p and q of $n/2$ bits; what is the expected value of the number of MSBs of $\sqrt{p}$ and $\sqrt{q}$ that are identical?

From our theoretical estimation, the expected value is about 2.6, and it is almost independent of the bit-length of N. This implies that, for any two randomly selected prime numbers p and q of $n/2$ bits each, the first 2.6 MSBs of $\sqrt{p}$ and $\sqrt{q}$ are identical on average. Consequently, according to (40), the size of $D_p - D_q$ is expected to be $2 \times (n/4 - 2.6) = n/2 - 5.2$ bits, which increases linearly with the bit-length of N.

4.2.2. Experimental Results. Table 1 shows the experimental results for the index t in EPF. Suppose that p and q are two randomly generated prime numbers of $n/2$ bits each; we then compute $\log_2(D_p - D_q)$, $\log_2(h_t)$, and $\log_2(h_{t+1})$, which denote the bit-lengths of $D_p - D_q$, h_t, and h_{t+1}, respectively. Each block in the table is evaluated from the average value of 1000 experimental instances. As can be observed from the first row, the bit-length of $D_p - D_q$ is approximately equal to $(n/2 - 7)$ bits long for all n and is greater than that of h_t by at least 1 bit on average. This result is slightly different from the result in the previous version at *ACNS'07* [23] due to the reason of using different samples in the experiments. Note that in this paper we implement EPF with uniformly distributed samples which are more objective. Moreover, the values of $\log_2(D_p - D_q)$ in Table 1 are slightly smaller than the theoretical estimation $n/2 - 5.2$ bits; the reason may be that

Table 1: The improvement of EPF on $p + q$, where p and q are balanced.

n	512	1024	2048
$\log_2(D_p - D_q)$	248.476	504.626	1016.551
t (in average)	146.229	295.772	594.103
$\log_2(h_t)$	247.161	503.04	1015.201
$\log_2(h_{t+1})$	250.12	506.21	1018.14

Table 2: The improvement of EPF on $p + q$, where p and q are balanced.

Balanced Modulus $N = pq$	$n = 512$	$n = 1024$	$n = 2048$
$\log_2((p+q) - 2\sqrt{N})$	248.476	504.626	1016.551
$\log_2((p+q) - (p_E + q_E))$	247.185	503.294	1015.248

we ignore the usage of prime-counting function $\pi(\cdot)$ in the calculation. However, the values in Table 1 actually increase linearly with the bit-length of N.

In EPF, we simply estimate the value of $D_p - D_q$, which is, however, smaller than the actual value. On the other hand, up to now, there is no theory to justify the difference between the bit-lengths of h_t and $D_p - D_q$; in fact, this would be an interesting subject of inquiry.

4.3. Accuracy and Further Improvement. We demonstrate the accuracy of EPF in Table 2. Each entry in the table is the data averaged over 1000 samples. The first row shows the difference of the bit-length between $p+q$ and its estimation by using $2\sqrt{N}$. The second row shows the difference of the bit-length between $p + q$ and its estimation by using EPF. As can be seen in Table 2, using $p_E + q_E$ as the estimation is more accurate than using $2\sqrt{N}$ at least one bit on average. This result shows that EPF is better than the traditional estimation method.

To further raise the accuracy rate of EPF, we may employ the properties of continued fractions. According to Theorem 2, we know that

$$h_{t+1} = a_t h_t + h_{t-1}, \qquad k_{t+1} = a_t k_t + k_{t-1}, \tag{41}$$

where a_t is the tth component of the continued fraction expression of $1/\sqrt{N}$ (see Definition in Section 2.2). Consequently, for any real number $\lambda \in [1, a_t]$, we have

$$h_t < \lambda h_t + h_{t-1} < h_{t+1}, \qquad k_t < \lambda k_t + k_{t-1} < k_{t+1}. \tag{42}$$

Since $D_p - D_q$ and $D_p \cdot D_q$ are also in the intervals (h_t, h_{t+1}) and (k_t, k_{t+1}), respectively, $\lambda h_t + h_{t-1}$ and $\lambda k_t + k_{t-1}$ might be better estimations of $D_p - D_q$ and $D_p \cdot D_q$. Hence, an interesting question would be how to find a suitable value of λ that yields better estimations of $D_p - D_q$ and $D_p \cdot D_q$. Note that, from the properties of continued fractions, we have

$$\begin{aligned} &\frac{h_{t+1}}{k_{t+1}} > \frac{1}{\sqrt{N}} > \frac{h_t}{k_t} \quad \text{if } t \text{ is old,} \\ &\frac{h_{t+1}}{k_{t+1}} < \frac{1}{\sqrt{N}} < \frac{h_t}{k_t} \quad \text{if } t \text{ is even.} \end{aligned} \tag{43}$$

Equation (43) implies that there exists an irrational number λ_1, such that

$$\frac{\lambda_1 h_t + h_{t-1}}{\lambda_1 k_t + k_{t-1}} = \frac{1}{\sqrt{N}}. \tag{44}$$

To find an appropriate number λ, one method could be to choose λ, which is very close to λ_1, which might yield better estimations of $D_p - D_q$ and $D_p \cdot D_q$. However, we leave this concept as the subject of future work on EPF.

5. Applying EPF to Reduce the Cost of Exhaustive Search to $2r-6$ Bits

In this section, we apply EPF proposed in Section 4 to further reduce the cost of exhaustive search.

From the results of Section 3, the security boundary of RSA-Small-d depends on the known MSBs of $p + q$. In EPF, the experimental results show that the 1st to the 8th MSB of $p+q$, denoted as $\mathrm{MSB}_{1\sim 8}(p+q)$, can be correctly determined with high probability (see Table 2). Consequently, setting $p_E + q_E = 2^{(n/2+1)-8}A_1 + A_2$, where $A_2 < 2^{n/2-7}$, then

$$(A_1)_2 = \mathrm{MSB}_{1\sim 8}(p+q), \tag{45}$$

where $(A_1)_2$ denotes the binary representation of A_1. Setting $\Lambda = (p+q) - (p_E + q_E)$, (45) also shows that Λ is about $(n/2 + 1) - 8$ bits long. Hence, representing (22) according to the bit-length of the items, Λ, d, k, and $\varphi(N)$ yields

$$\left(\left(\frac{n}{2} + 1\right) - 8\right) + 1 + 2\left(\frac{n}{4} + r\right) < n. \tag{46}$$

Moreover, by performing an exhaustive search for s bits after the 8th MSB of $p + q$, that is, $\mathrm{MSB}_{9\sim 8+s}(p+q)$, we can further reduce the size of Λ to $(n/2+1) - (8+s)$ bits. This implies that the 1st to the $(8+s)$th MSB of $p+q$ can be correctly determined and the size of Λ is reduced to $(n/2 + 1) - (8 + s)$ bits. Hence, (46) is revised to

$$\left(\frac{n}{2} + 1\right) - (8 + s) + 1 + 2\left(\frac{n}{4} + r\right) < n, \tag{47}$$

which is simplified to

$$2r - 6 < s. \tag{48}$$

Equation (48) is the improved result when applying EPF to the method presented in Section 3. As a conclusion, extending Wiener's boundary by r bits requires only an exhaustive search for $2r - 6$ bits, which results in a lower computational cost than that with Verheul and van Tilborg's extension. We summarize the improvements in each type of attack in Table 3.

With the progress of technology, the ability of machines to perform exhaustive searches will only increase. Figure 1 shows the relations between the security boundaries of the extensions of the Wiener attack and machines with different computational abilities. The symbol s denotes the required number of bits for an exhaustive search to extend Wiener's boundary, and the symbol $|d|$ denotes the upper

Table 3: The improvement between each attack.

Attacks	Boundary	Complexity
Wiener Attack	$d < N^{1/4}$	Polynomial time
V-T Extension	$d < N^{1/4}2^r$	Exhaustive search for U and V of $2r + 8$ bits (see (13))
Proposed Improvement (Step 1)	$d < N^{1/4}2^r$	Exhaustive search for $2r + 2$ bits (see (24))
Applying EPF (Step 2)	$d < N^{1/4}2^r$	Exhaustive search for $2r - 6$ bits (see (48))

bound of the insecure private exponent. In terms of the current computational capabilities, an exhaustive search for 64 bits is feasible. Hence, the lines **L1**, **L2** and **L3** yield the improvements of 28 bits, 31 bits, and 35 bits, respectively, over Wiener's boundary. The boundaries of the extensions of the Wiener attack (see V-T. Ext., Ext. W., and EPF in Figure 1) can be raised to 284 bits, 287 bits, and 291 bits, respectively, when the RSA modulus N is 1024 bits long. Furthermore, if an exhaustive search for 80 bits is feasible, the upper bound of the extension of the Wiener attack through EPF is raised to $N^{1/4}2^{43}$, which is 299 bits when N is 1024 bits long (see **L3**: EPF). This result is comparable to the boundary of the lattice attack proposed by Boneh and Durfee [19], which has a best upper bound, but heuristic, at the present. Note that there is no guaranty that a heuristic algorithm can output the solution. One may concern whether the assumption that an exhaustive search for 80 bits is feasible or not. In the opinion of current development, it will not be a difficult task to achieve such computational capability in the near future. According to Moore's Law, computers will double in speed approximately every 18 months, which further supports our assumption. Moreover, paralleling techniques and special-purpose machines can help in speeding-up the computation.

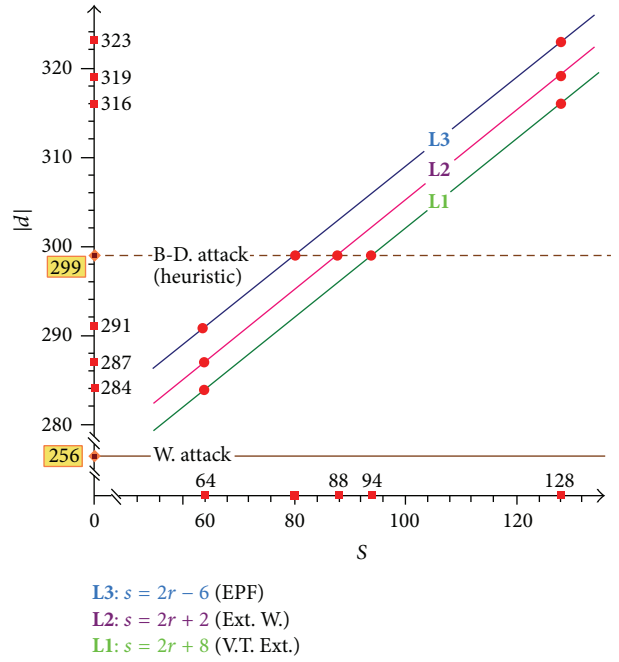

Figure 1: The boundaries of the extensions of the Wiener attack under different computational capabilities, where 256 and 299 are the boundaries of the Wiener attack (W. Attack) and Boneh and Durfee's attack (B-D. Attack), respectively. **L1**, **L2**, and **L3** denote the boundaries of Verheul and Tilborg's extension (V-T. Ext.) (see [20]), the extension of the Wiener attack (Step 1) (Ext. W.) (see (24)), and the extension of the Wiener attack through EPF (EPF) (see (48)).

6. Conclusion and Future Works

With the rapid growth of different network environments such as wireless sensor networks [24–27], security is normally the most concerned issue. In this paper, we propose a method, called EPF, to estimate the prime factors of an RSA modulus. With EPF, the cost of exhaustive search can further reduce to $2r - 6$ bits. It means that the cost is 2^{14} times faster than Verheul and van Tilborg's result and the security boundary is extended 7 bits. It should be noted that their method for an exhaustive search is heuristic since this method is based on the results of distribution of small partial quotient in the continued fraction expansions.

An interesting problem in EPF is whether there exists a deterministic algorithm for finding an index t satisfying $h_t < D_p - D_q < h_{t+1}$. In this paper, we use the theoretical estimation to determine the index t. The success rate is 85.1% according to our experiments. Now, another question arises—how to increase the success rate of the process of finding the index t when the deterministic algorithm is not developed. In addition, the other researchable question is how to improve the accuracy rate of MSBs of $p_E + q_E$, which brings a greater contributive effort of EPF.

We should point out that EPF can be applied to Dujella's refinement [14] and the generalized Wiener attack [18]. Moreover, we foresee that EPF could be applied to other cryptogrammic aspects, especially to the attacks for cryptosystems based on the integer factorization problem (IFP). For example, the lattice technique is commonly used for the cryptanalysis of RSA [17, 28–30] or for the attacks on RSA with small exponents [15, 18, 19, 21, 22, 31, 32]. We expect EPF to be a supportive tool for assisting the lattice technique to increase the effort on the cryptanalysis of RSA. As a conclusion, we would like to point out that with the continuous improvements in computational capability, the security levels are expected to be higher with the assistance of EPF, and the security analysis should be considered more carefully.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions, which certainly led to improvements of this paper. Chien-Ming Chen was partially supported by the Shenzhen Peacock Project, China, under Contract no. KQC201109020055A and the Shenzhen Strategic Emerging Industries Program under Grant no. ZDSY20120613125016389. Hung-Min Sun was partially supported by the National Science Council, Taiwan, under Grant NSC 100-2628-E-007-018-MY3.

References

[1] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[2] C. Patsakis, "Number theoretic SETUPs for RSA like factoring based algorithms," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 3, no. 2, pp. 191–204, 2012.

[3] Q. Kong, P. Li, and Y. Ma, "On the feasibility and security of image secret sharing scheme to identify cheaters," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 4, no. 4, pp. 2073–4212, 2013.

[4] N. Peng, G. Luo, K. Qin, and A. Chen, "Query-biased preview over outsourced and encrypted data," *The Scientific World Journal*, vol. 2013, Article ID 860621, 13 pages, 2013.

[5] H. Lenstra Jr., "Factoring integers with elliptic curves," *Annals of Mathematics*, vol. 126, no. 3, pp. 649–673, 1987.

[6] J. Pollard, "Theorems on factorization and primality testing," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 76, no. 3, pp. 521–528, 1974.

[7] D. Boneh and H. Shacham, "Fast variants of RSA," *CryptoBytes*, vol. 5, no. 1, pp. 1–9, 2002.

[8] S. Galbraith, C. Heneghan, and J. McKee, "Tunable balancing of RSA," in *Information Security and Privacy*, vol. 3574 of *Lecture Notes in Computer Science*, pp. 280–292, Springer, Berlin, Germany, 2005.

[9] M. Hinek, "Another look at small RSA exponents," in *Topics in Cryptology-CT-RSA 2006*, vol. 3860 of *Lecture Notes in Computer Science*, pp. 82–98, Springer, Berlin, Germany, 2006.

[10] H. Sun, W. Yang, and C. Laih, "On the design of RSA with short secret exponent," in *Advances in Cryptology-ASIACRYPT '99*, vol. 1716 of *Lecture Notes in Computer Science*, pp. 150–164, Springer, Berlin, Germany, 1999.

[11] H. Sun and C. Yang, "RSA with balanced short exponents and its application to entity authentication," in *Public Key Cryptography-PKC 2005*, vol. 3386 of *Lecture Notes in Computer Science*, pp. 199–215, Springer, Berlin, Germany, 2005.

[12] S. Vanstone and R. Zuccherato, "Short RSA keys and their generation," *Journal of Cryptology*, vol. 8, no. 2, pp. 101–114, 1995.

[13] D. Boneh, R. Rivest, A. Shamir et al., "Twenty years of attacks on the RSA cryptosystem," *Notices of the American Mathematical Society*, vol. 46, no. 2, pp. 203–213, 1999.

[14] A. Dujella, "Continued fractions and RSA with small secret exponent," *Tatra Mountains Mathematical Publications*, vol. 29, pp. 101–112, 2004.

[15] E. Jochemsz and B. de Weger, "A partial key exposure attack on RSA using a 2-dimensional lattice," in *Information Security*, vol. 4176 of *Lecture Notes in Computer Science*, pp. 203–216, Springer, Berlin, Germany, 2006.

[16] M. J. Wiener, "Cryptanalysis of short RSA secret exponents," *IEEE Transactions on Information Theory*, vol. 36, no. 3, pp. 553–558, 1990.

[17] B. de Weger, "Cryptanalysis of RSA with small prime difference," *Applicable Algebra in Engineering, Communications and Computing*, vol. 13, no. 1, pp. 17–28, 2002.

[18] J. Blömer and A. May, "A generalized Wiener attack on RSA," in *Public Key Cryptography-PKC 2004*, vol. 2947 of *Lecture Notes in Computer Science*, pp. 1–13, Springer, Berlin, Germany, 2004.

[19] D. Boneh and G. Durfee, "Cryptanalysis of RSA with private key d less than $N^{0.292}$," in *Advances in Cryptology-EUROCRYPT '99*, vol. 1592 of *Lecture Notes in Computer Science*, pp. 1–11, Springer, Berlin, Germany, 1999.

[20] E. Verheul and H. van Tilborg, "Cryptanalysis of "less short" RSA secret exponents," *Applicable Algebra in Engineering, Communications and Computing*, vol. 8, no. 5, pp. 425–435, 1997.

[21] D. Coppersmith, M. Franklin, J. Patarin, and M. Reiter, "Low-exponent RSA with related messages," in *Advances in Cryptology-EUROCRYPT '96*, vol. 1070 of *Lecture Notes in Computer Science*, pp. 1–9, Springer, Berlin, Germany, 1996.

[22] D. Boneh and G. Durfee, "Cryptanalysis of RSA with private key d less than d less than $N^{0.292}$," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1339–1349, 2000.

[23] H. M. Sun, M. E. Wu, and Y. H. Chen, "Estimating the prime-factors of an rsa modulus and an extension of the wiener attack," in *Applied Cryptography and Network Security*, vol. 4521 of *Lecture Notes in Computer Science*, pp. 116–128, Springer, Berlin, Germany, 2007.

[24] C. M. Chen, Y. H. Lin, Y. H. Chen, and H. M. Sun, "SASHIMI: secure aggregation via successively hierarchical inspecting of message integrity on WSN," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 4, no. 1, pp. 57–72, 2013.

[25] C. M. Chen, Y. H. Lin, Y. C. Lin, and H. M. Sun, "RCDA: recoverable concealed data aggregation for data integrity in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 4, pp. 727–734, 2012.

[26] H.-M. Sun, C.-M. Chen, and Y.-C. Hsiao, "An efficient countermeasure to the selective forwarding attack in wireless sensor networks," in *Proceedings of the IEEE Region 10 Conference (TENCON '07)*, pp. 1–4, IEEE, Taipei, Taiwan, November 2007.

[27] K. Wei-Chi, C. Chien-Ming, and L. Hui-Lung, "Cryptanalysis of a variant of Peyravian-Zunic's password authentication scheme," *IEICE Transactions on Communications*, vol. 86, no. 5, pp. 1682–1684, 2003.

[28] D. Coppersmith, "Finding a small root of a bivariate integer equation; factoring with high bits known," in *Advances in Cryptology-EUROCRYPT '96*, vol. 1070 of *Lecture Notes in Computer Science*, pp. 178–189, Springer, Berlin, Germany, 1996.

[29] D. Coppersmith, "Finding a small root of a univariate modular equation," in *Advances in Cryptology-EUROCRYPT '96*, vol. 1070 of *Lecture Notes in Computer Science*, pp. 155–165, Springer, Berlin, Germany, 1996.

[30] H. Sun, M. Wu, W. Ting, and M. Hinek, "Dual RSA and its security analysis," *IEEE Transactions on Information Theory*, vol. 53, no. 8, pp. 2922–2933, 2007.

[31] D. Bleichenbacher and A. May, "New attacks on RSA with small secret CRT-exponents," in *Public Key Cryptography-PKC 2006*, vol. 3958 of *Lecture Notes in Computer Science*, pp. 1–13, Springer, Berlin, Germany, 2006.

[32] D. Boneh, G. Durfee, and Y. Frankel, "An attack on RSA given a small fraction of the private key bits," in *Advances in Cryptology-ASIACRYPT '98*, vol. 1514 of *Lecture Notes in Computer Science*, pp. 25–34, Springer, Berlin, Germany, 1998.