Future Generation Computer Systems 30 (2014) 162–168

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

On the improvement of Fermat factorization using a continued fraction technique

Mu-En Wu [a], Raylin Tso [b], Hung-Min Sun [c,*]

[a] Institute of Information Science, Academia Sinica, Taipei, Taiwan
[b] Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan
[c] Department of Computer Science, National Chengchi University, Taipei, Taiwan

HIGHLIGHTS

- Proposing a secure and efficient privacy preserving RFID authentication protocol.
- Using an RFID back-end server with cloud database to reduce the search complexity as well as data consistency.
- Withstanding desynchronizing attacks and tracking attacks.
- Providing scalability with $O(\log N)$ search complexity.

ARTICLE INFO

Article history:
Received 5 January 2013
Received in revised form
3 May 2013
Accepted 9 June 2013
Available online 9 July 2013

Keywords:
Integer factorization problem
IFP
Fermat's algorithm
Continued fraction
Estimated prime factor

ABSTRACT

Although Integer Factorization Problem (IFP) is one of the most difficult problems in the world due to the limited computational capability, there exist some vulnerable integers which are factorable by the development of cloud computing. For example, given an integer $N = pq$, which is a product of two primes, it is hard to determine the prime factors p and q efficiently. However, for the suitable size of a number N, Fermat's algorithm may be one of the simplest method for solving it. In this paper, a method called *EPF* for estimating the prime factors of a composite number is proposed. We use the technique of continued fractions to output two integers, $p_E + q_E$ and $p_E \cdot q_E$, which are close to $p + q$ and $p \cdot q$, respectively. Furthermore, we show that *EPF* can be adopted to reduce the loop count in Fermat's algorithm before factoring a composite number. The effect depends on the size of the prime factor. We believe that there are still other applications as well wherein *EPF* can be used.

1. Introduction

Lots of cryptosystems are based on the hardness of the integer factorization problem (IFP), which remains a well-studied problem [1,2] in the past 30 years. Currently, it is suggested that the bit-length of a composite number N should be at least 1024 to be considered secure. Using the best-known factoring algorithms, the expected workload of factoring a 1024-bit number $N = pq$ is 2^{80} which is currently believed to be infeasible. In this paper we develop an approach, called "Estimated Prime Factor (*EPF*)", to estimate $p + q$, and then 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 efficiently determined. This result is more accurate than the traditional estimation, which estimates $p + q$ by $2\sqrt{N}$.

Furthermore, we show that *EPF* [3] can be adopted to reset the initial values of Fermat's algorithm in order to reduce the loop count. The reduced amount depends on the bit-lengths of the prime factors. Taking 16-bit p and q for experiment, the new setting reduces approximately 62% of the loop count when running Fermat's algorithm and thus enhances the efficiency by about 2.45 times.

Other than the above mentioned application, *EPF* may provide a basis for aiding cryptanalysis, such as lattice attacks [4–15], integer factoring algorithms, and so on. For example, according to the cryptanalysis proposed by Weger [15], the boundary of Boneh and Durfee's lattice attack will be extended if the difference between p and q decreases. This result suggests that *EPF* afford some advantages when applied to the lattice attack.

The remainder of this paper is organized as follows: Section 2 presents the preliminaries of this paper. In Section 3, we propose the *EPF* approach for estimating the prime factors of an RSA modulus. Next, an application of *EPF* is adopted to improve Fermat's Factorization in Section 4. Finally, we present our conclusions

* Corresponding author. Tel.: +886 35742968.
E-mail addresses: mnasia1@gmail.com (M.-E. Wu), raylin@cs.nccu.edu.tw (R. Tso), hmsun@cs.nthu.edu.tw, bensonbessie@gmail.com (H.-M. Sun).

0167-739X/$ – see front matter
http://dx.doi.org/10.1016/j.future.2013.06.008

in Section 5, where making the conclusion, along with recommendations for future work.

2. Preliminary: Fermat's algorithm

Fermat's algorithm is a kind of methods for solving the integer factorization problem. It is well-suited for use with small computers to factor small composite numbers [16]. Without loss of generality, we suppose that the input of Fermat's algorithm is a composite integer $N = pq$ with odd numbers p and q. Note that here p and q may be composite numbers. Fermat's factoring method looks for two integers u and v such that $4N = u^2 - v^2$. Since $4N$ can be rewritten as the difference of two perfect squares, which is

$$4N = 4pq = (p+q)^2 - (p-q)^2, \tag{1}$$

we assume that $u := p+q$ and $v := p-q$ from (1). In Fermat's algorithm, setting $r = u^2 - v^2 - 4N$, we would like to find a solution (u, v) satisfying $r = u^2 - v^2 - 4N = 0$. *i.e.*,

$$4N = u^2 - v^2 = (u+v) \cdot (u-v).$$

If such (u, v) is found, then p and q can be computed by $p = \frac{u+v}{2}$ and $q = \frac{u-v}{2}$ immediately. Fermat's algorithm performs an exhaustive search to find the solution (u, v). Setting the initial values of u and v by

$$u \leftarrow 2\lceil\sqrt{N}\rceil \quad \text{and} \quad v \leftarrow 0,$$

two cases making $r \neq 0$ are considered. First, if a pair (u, v) makes $r > 0$, *i.e.*, $r = u^2 - v^2 - 4N > 0$, then it implies that v should be set larger. The next possible value of v is to set $v \leftarrow v + 2$. Note that we should not reset v by $v \leftarrow v + 1$ because $p - q$ is an even number. After resetting $v \leftarrow v + 2$, r changes to

$$r \leftarrow r - (4v + 4), \tag{2}$$

where $4v + 4$ comes from the difference between the new value of v and the old one. That is,

$$(v+2)^2 - v^2 = 4v + 4.$$

Secondly, if a pair (u, v) makes $r < 0$, *i.e.*, $r = u^2 - v^2 - 4N < 0$, then it implies u should be set larger. We set $u \leftarrow u + 2$ and r changes to

$$r \leftarrow r + (4u + 4), \tag{3}$$

where $4u + 4$ comes from the difference between the new value of u and the old one. That is,

$$(u+2)^2 - u^2 = 4u + 4.$$

Repeating the processes of (2) and (3) until $r = 0$, it completes the factorization of N. We give the pseudo code of Fermat's algorithm in the following. The detail can be referenced in [17].

Fermat's algorithm	
INITIALIZE:	READ $N = pq$
	$u \leftarrow 2\lceil\sqrt{N}\rceil$ and $v \leftarrow 0$
	$r \leftarrow u^2 - v^2 - 4N$
X_LOOP:	WHILE $r \neq 0$
	IF $r > 0$ THEN CALL Y_LOOP
	IF $r < 0$ THEN DO
	$r \leftarrow r + (4u + 4)$ and
	$u \leftarrow u + 2$
Y_LOOP:	WHILE $r > 0$ DO
	$r \leftarrow r - (4v + 4)$ and
	$v \leftarrow v + 2$
	RETURN
TERMINATE:	$p \leftarrow \frac{u+v}{2}$ and $q \leftarrow \frac{u-v}{2}$
	WRITE p, q

In Section 4, we revise the part of "INITIALIZE" in Fermat's algorithm. Through *EPF*, the new initial values u and v are adopted, instead of setting $u \leftarrow 2\lceil\sqrt{N}\rceil$ and $v \leftarrow 0$, to enhance the algorithm efficiency. We discuss the improvement later and following are the notations which will be used in the paper.

Category	Notation
EPF	p_E: the estimation of p.
	q_E: the estimation of q.
	D_p: the distance between $\sqrt{N}$ and p.
	D_q: the distance between q and $\sqrt{N}$.
	t: the index of continued fraction satisfying $h_t < D_p - D_q < h_{t+1}$.

3. An approach (*EPF*) to estimate $p + q$

In this section, a novel approach called *EPF*, which is used to estimate the prime factors of a composite number N, is proposed. *EPF* is an abbreviation for "estimated prime factor". Two cases of moduli N for *EPF* are considered: the balanced modulus and the unbalanced modulus. Firstly, we show *EPF* when N is balanced.

3.1. Balanced modulus

In the case of the balanced modulus $N = pq$, without loss of generality, we assume that $q < p < 2q$. Denote D_p and D_q as the distances between $\sqrt{N}$ & p, and q & $\sqrt{N}$ respectively. That is,

$$p = \sqrt{N} + D_p \quad \text{and} \quad q = \sqrt{N} - D_q. \tag{4}$$

Applying (4) to $N = pq$ yields

$$\begin{aligned} N = p \cdot q &= (\sqrt{N} + D_p) \cdot (\sqrt{N} - D_q) \\ &= N + \sqrt{N}(D_p - D_q) - D_p D_q. \end{aligned} \tag{5}$$

Eliminating N in both sides of (5) yields $D_p D_q = \sqrt{N}(D_p - D_q)$, which leads to

$$\frac{1}{\sqrt{N}} = \frac{D_p - D_q}{D_p D_q}. \tag{6}$$

Eq. (6) is quite interesting because the irrational fraction $\frac{1}{\sqrt{N}}$ reveals partial information of $D_p - D_q$ and $D_p D_q$. Note that with $D_p - D_q$ and $D_p D_q$ we can compute $D_p + D_q$ by

$$(D_p + D_q)^2 = (D_p - D_q)^2 + 4D_p D_q, \tag{7}$$

and solve D_p and D_q as follows:

$$D_p = \frac{D_p + D_q}{2} + \frac{D_p - D_q}{2} \quad \text{and}$$
$$D_q = \frac{D_p + D_q}{2} - \frac{D_p - D_q}{2}.$$

Now we use continued fractions to construct a rational sequence to approximate $\frac{1}{\sqrt{N}}$. Suppose the ith convergent of the continued fraction expansion of $\frac{1}{\sqrt{N}}$ is $\frac{h_i}{k_i}$. From the property of continued fraction, we know $\frac{h_i}{k_i} \to \frac{1}{\sqrt{N}}$, as $i \to \infty$. Since the sizes of h_i and k_i grow with the increase of the index i, there exists an index t such that

$$h_t < D_p - D_q < h_{t+1}. \tag{8}$$

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,

$$h_t \approx D_p - D_q \quad \text{and} \quad k_t \approx D_p D_q. \tag{9}$$

From (7), $D_p + D_q$ is estimated as

$$D_p + D_q \approx \sqrt{h_t^2 + 4k_t} \tag{10}$$