Modified Prime Number Factorization Algorithm (MPFA) For RSA Public Key Encryption

Kuldeep Singh, Rajesh Verma, Ritika Chehal

Abstract*— The Public key encryption security such as RSA scheme relied on the integer factoring problem. The security of RSA algorithm based on positive integer N, which is the product of two prime numbers, the factorization of N is very intricate. In this paper a factorization method is proposed, which is used to obtain the factor of positive integer N. The present work focuses on factorization of all trivial and nontrivial integer numbers as per Fermat method and requires fewer steps for factorization process of RSA modulus N. By experimental results it has been shown that factorization speed becomes increasing as compare to traditional Trial Division method, Fermat Factorization method, Brent's Factorization method and Pollard Rho Factorization method.

***Index Terms*— Factorization Problem, MPFA, Public Key Cryptography, RSA Scheme..**

I. INTRODUCTION

The concept of public-key cryptography evolved from an attempt to attack one of the most difficult problems associated with non-conventional encryption, the first problem is that of key distribution as we have seen, key distribution under non-conventional encryption requires that, to communicates either first already share a key, which somehow has been distributed to them, or second have the use of a key distribution center. Public key cryptography is one of the mathematical applications that are valuable in sending information via insecure channel. RSA algorithm is a public key encryption algorithm. RSA has become most popular cryptosystem in the world because of its simplicity. The principle of RSA algorithm is that: according to number theory, it is easy to finds two big prime number, but the factorization of the product of two big prime numbers is very difficult. The difficulty of computing the roots of *N*, where *N* is the product of two large unknown primes, it is widely believed to be secure for large enough *N*. Since RSA can also be broken by factoring *N*, the security of RSA is often based on the integer factorization problem [1]. A method for factoring algorithm (specially designed) for semiprimes based on new mathematical ideas. Since this method is relatively simple and scalable, it can be suitable for parallel processing. A new algorithm to attacks the RSA scheme. But the main condition of this algorithm is that to break RSA modulus firstly we should have public key and modulus n. On the basis of this public key (e, n) proposed algorithm disclose the private key. Introducing some weak points of RSA algorithm and proposed a method for secure public key.

Manuscript received September 02, 2012.

Kuldeep Singh, Department of Computer Science & Engineering,Jind Institute of Engineering and Technology, Jind, India.

Dr. Rajesh Kr. Verma, Professor & Head, Deptt. Of CSE, Kurukshetra Institute of Technology & Management, Kurukshetra(Haryana),India.

Ritika Chehal, Department of Computer Science & Engineering,Jind Institute of Engineering and Technology, Jind, India.

The main concept of this paper is for encryption *(e, n)* change every time public key for encryption. In this paper, we are focusing on generation of a private key because the generation of private key is dependent Euler's totient function $\Phi(N)=(p-1)(q-1)$, p and q is prime factors of $N=p*q$, $p!=q$, private key $d=e^{-}\bmod(\Phi(N))$, so we are concluding that if we can find out the prime factors of n, then we can easily generate private key. In this paper a Modified Prime Number Factorization Algorithm (MPFA) method based on Fermat method [2] is proposed. This method can factorize only product of two prime numbers. By using this method; we can factorize rapidly all positive integer number *N*, which is the product of two prime numbers, JAVA environment is used for various analyses.

Use either SI (MKS) or CGS as primary units. (SI units are strongly encouraged.) English units may be used as secondary units (in parentheses). This applies to papers in data storage. For example, write "15 Gb/cm^2 (100 Gb/in^2)." An exception is when English units are used as identifiers in trade, such as "3½ in disk drive." Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity in an equation.

The SI unit for magnetic field strength *H* is A/m. However, if you wish to use units of T, either refer to magnetic flux density *B* or magnetic field strength symbolized as $\mu_0 H$. Use the center dot to separate compound units, e.g., "A·m^2."

II. RELATED WORK

J. Carlo et al, [3] have proposed a method for factoring algorithm is: gcd[N, $(k.N')+\Delta$] and gcd[N, $(k.N')+\Delta$] result in nontrivial factors of N for different values of Δ where N' is the reverse of N and k is a positive integer ranging from one to infinity.

Sattar J Aboud [4] proposed a method breaking the RSA scheme based on the knowing public key *(e,n)*. This method will work efficiently if the decryption key d is small. Therefore, it is possible to factor the modulus *N*.

L. Scripcariu, M.D. Frunze [5] introduced some weak points of RSA algorithm and proposed a method for secure public key. The main concept of this paper is for encryption *(e, n)* change every time public key for encryption.

III. RSA SCHEME

In 1977, RSA developed a public key cryptosystem that is based on the difficulty of integer factoring [1]. The RSA public key encryption scheme is the first example of a provable secure public key encryption scheme against chosen message chosen attacks [6].

The RSA scheme is as follows [1]: Key generation algorithm, to generate the keys entity *A* must do the following:
1. Randomly and secretly select two large prime numbers *p* and *q*.
2. Compute the *N = p*q.*
3. Compute *Φ (N) = (p-1) (q-1).*
4. 4. Select random integer *e, 1<e<N*, where gcd (e, *Φ* (n))=1
5. Compute the secret exponent *d, 1 < d <* phi, such that *ed* ≡ 1(mod (*Φ (N)*)).
6. The public key is *(N, e)* and the private key is *(N,d).*
7. Keep all the values d, p, q and *Φ (N)* secret.

- *N* is known as the *modulus*.
- e is known as the public exponent or encryption exponent or just the exponent.
- d is known as the secret exponent or decryption exponent

Public key encryption algorithm:
Entity *A* encrypt a message *M* for entity *B* which entity decrypt. Entity A should do the following:

- Obtain entity *A*'s public key(*N,e*)
- Represent the message *M* as an integer in the interval [0….*N*-1].
- Compute $C = M^e \bmod N$.
- Send the encrypted message *C* to entity *B*.

Decryption: To recover the message m from the cipher text *C*. Entity *B* performs the following:

- Obtain the cipher text *C* from entity *A*.
- Recover the message $M = C^d \bmod n$.

Trial Division: Trial division is the simplest algorithm for factoring an integer. Assume that S and T are nontrivial factors of *N* such that *ST = N* and *S < T*. To perform the trial division algorithm, one simply checks whether *N s* | for S =2,……,floor(√*N*) . When such a divisor S is found, then *T = N / S* is also a factor, and a factorization has been found for *N*. The upper bound of *S* ≤ floor (√*N*) is provided by the following theorem:

If *N* has nontrivial factors *S, T* with *ST = N* and *S*≤ *T*, then *S*≤ √*N*

Fermat factorization algorithm in the 1600s [5a] was discovered by mathematician Pierre de Fermat. Fermat factorization rewrites a composite number *N* as the difference of squares: $N = x^2 - y^2$, this difference of squares leads immediately to the factorization of *N*: Assume that *s* and *t* are nontrivial odd factors of *N* such that *st = N* and *s < t*. We can find *x* and *y* such that *s =(x - y)* and *t = (x +y).*

Pollard rho Factorization [7] method is a probabilistic method for factoring a composite number *N* by iterating a polynomial modulo *N*. The method was published by J.M. Pollard in 1975. Suppose we construct the sequence: $x_0 \equiv 2 \pmod N$, $x_{n+1} \equiv x^2_n + 1 \pmod N$. This sequence will eventually become periodic. It can be shown that the length of the cycle is less than or equal to *N* by a proof by contradiction: assume that the length *L* of the cycle is greater than *N*, however we have only *N* distinct X_n values in our cycle of length *L > N*, so there must exist two X_n values are congruent, and these can be identified as the starting points of a cycle with length less than or equal to *N*. Probabilistic arguments show that the expected time for this sequence (mod *N*) to fall into a cycle and expected length of the cycle are both proportional to *N* , for almost all *N* [8]. Other initial values and iterative functions often have similar behaviour under iteration, but the function $f(n) = x^2_n+1$ has been found to work well in practice for factorization. Assume that *s* and *t* are nontrivial factors of *N* such that *st =N* and *s* ≤ *t*. Now suppose that we have found nonnegative integers *i, j* with *i < j* such that $x_i \equiv x_j \pmod s$ but $x_i \neq x_j \pmod N$. Since *s* | $(x_i - x_j)$, and *s* | *N*, we have that *s* | $gcd(x_i - x_j , N)$. By assumption $s \geq 2$, thus $gcd(x_i - x_j , N) \geq 2$. By definition we know $gcd(x_i - x_j , N) \mid N$. However, we have that $N \neq gcd(x_i - x_j , N)$, and thus that $N \neq gcd(x_i - x_j , N)$. So we have $N \neq gcd(x_i - x_j , N)$, $gcd(x_i - x_j , N) > 1$, *and* $gcd(x_i - x_j , N) \neq N$. Therefore $gcd(x_i - x_j , N)$ is a nontrivial factor of *N*.

Example1: Consider the Pollard rho algorithm for *N* = 21 =7*3. The sequence of *Xn* values generated by the algorithm is $X_0 = 2$, $X_1 = 5$, $X_2 = 5, \ldots\ldots, n \geq 1$, *If* $n \geq 1$, $X_{2n} - X_n = 0$. The algorithm at each step for *n = 1,2,…..* computes gcd $(X_{2n} – X_n, N)$ = gcd *(0, N) =N*. By this example, it's shown that the Pollard rho algorithm is a probabilistic, and may not finish, even for small values of *N*.

IV. PROPOSED METHOD

A lot of algorithm has been proposed regarding factorization, the Pollard rho algorithm [7], and Brent's method [9], are probabilistic, and may not finish, even for small values of *N*, but Trial division algorithm and proposed method can finish all trivial and nontrivial values of N, shown in Table 1. This method is not probabilistic. To break RSA in to two prime numbers we should have the product of that prime numbers is equal to N. Factorization of N is very difficult to find that prime number. MFF can factors of N, which is P and Q, are its respective prime factors. Various steps involved in the method are as follows:

1. B := floor ($N^{1/4}$*T) // where is T is tunable
   // constant; let T=1
2. for i := 2: B
3. if N% i = 0
   Return i, *N*/i
4. end for
5. k := 0
6. loop forever
7. k := k + 1
8. if k*T^3 > B
   Return (*N* is prime)
9. end if
10. if k is even
    m := 2; r := 1;
11. else
    m := 4; r := (k+N)%4;
12. end if
13. For all integers a with 0 ≤ a*a - 4*k*N ≤ B*B and with a % m = r
14. c := a*a-4*k*N
15. if isSquare(c)
    i. b := √c
    ii. P := gcd(a+b, N)
    iii. Q := N/P
    iv. return P,Q
16. end if
17. end loop

Example 2:
Let N=25
Decimal number = 2
Number of bits = 7

Lets factor := P,Q
Compute B := 2
Goto step 13
Compute a :=10
Compute c:=10*10-4*1*25 = 10
Compute b := 0
c is squre, therefore P = gcd(10+0, 25)=5
Q := 25/ 5 = 5

Factorization Method	For all numbers	Technique used
Trial Method	Can factorize	Division based
Fermat Factorization	Can factorize	$N= X^2 –Y^2$
Pollard Rho	Can't factorize	Periodic sequences
MPFA	Can factorize	Based on fermat method

Table 1: Comparison of some factorization methods vs MPFA method.

V. RESULTS

Table 2 shows the running steps taken to factorize the input *N* into its factors by different algorithms. From the table, it's shown that the number of steps increases with the increase in the digits of the factors. *Figure 1* shows a plot of time elapsed and number of digits for given number by using traditional Trial method, Fermat Factorization method and proposed MPFA method. The algorithm was executed using Java development kit and Intel(R) core 2 Quad CPU, 2.66 GHz, 4 GB of RAM. RSA 1024 can also be break by using above proposed method needs 64-bits compiler [10].

Decimal Digits in Prime Factors	Trial Division Factorization Algorithm	Fermat's Factorization Algorithm	Pollard Rho Factorization Algorithm	MPFA
1	1	1	1	1
2	30	5	5	7
3	300	60	20	40
4	3000	1000	50	100
5	30000	3000	150	100
6	300000	40000	400	200

Table 2: Comparison of digits in the Prime factors w.r.t. steps taken by different factorization methods

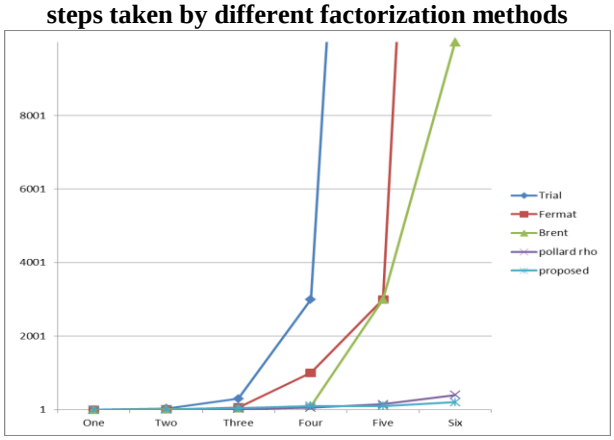

Figure 1: Number of steps taken *(y-axis)* vs Digits in prime factors *(x-axis)*

VI. CONCLUSION

In this paper an algorithm is proposed for RSA modulus factorization. The new algorithm aims to obtain the prime factors of modulus *N* in RSA algorithm. In this method we are dividing Fermat factorization method in two part first is one is, factorize number with respect floor function of square root of *N*, because we get maximum factors are neighbor to the that value, second is if we don't get positive integer value of square root (square root of *N*), then we sequence between ceil($\sqrt{N}$) to *N*. Shown in *Figure 1*, steps taken for prime factorization are decreasing as compare to the Fermat and trial division method. So we conclude, if we find prime factor of RSA modulo *N*, then we can generate private key and decrypt to the secret message. This algorithm is relatively simple and scalable. Proposed future work if we can remove process of decryption with independent of *N*, by using this method, we can increase security strength of RSA algorithm. MPFA method can be used for factorization of positive integer N, very helpful to generate results at efficient and faster rate.

REFERENCES

1. Rivest, R.; A. Shamir; L. Adleman. *"A Method for Obtaining Digital Signatures and Public-Key Cryptosystems"*, Communications of the ACM 21 (2): 120–126, doi: 10.1145/359340.359342, 1977
2. Bell, E. T. *"The Prince of Amateurs: Fermat."*, New York: Simon and Schuster, pp. 56-72, 1986.
3. João Carlos Leandro da Silva, *"Factoring Semi primes and Possible Implications"*, IEEE in Israel, 26th Convention, pp. 182-183, Nov. 2010.
4. Sattar J Aboud, *"An efficient method for attack RSA scheme"*, IEEE 2009.
5. L. Scripcariu, M.D. Frunza, *"A New Character Encryption Algorithm"*, Proceedings of the Intern. Conference on Microelectronics and Computer Science, Chisinau, (Republica Moldova), ICMCS 2005, pp. 83 - 86, Sept., 2005.
6. B. Schneier, Applied cryptography, second edition, NY: John Wiley & Sons, Inc., 1996.
7. J. Pollard, *"Monte Carlo methods for index computation (mod p)"*,Math. Comp., Vol. 32, pp.918-924, 1978.
8. R. P. Brent, *"An improved Monte Carlo factorization algorithm"*, BIT 20 (1980), 176-184. MR 82a:10007, Zbl 439.65001. rpb051.

AUTHORS PROFILE

Kuldeep Singh Kundu received first Master Degree as Master of Computer Application (M.C.A.) in 2006 from Chaudhary Devi Lal University, Sirsa, Haryana, India and Second Master Degree as M.Tech. in Computer Sciene and Engineering in 2008 from Guru Jambheshwer University of Science and Technology,Hisar, Haryana, India. He is presently working as Assistant Professor in Jind Institute of Engineering and Technology, Jind,India. His area of research include Routiong Protocols, Mobile Ad hoc Networks, Cryptograhic approcheses with network simulator as like GloMoSim and ns-2. he has already 7 publications in many conferences.

Dr. R.K Verma is working as Professor and Head in Deptt. of CSE at Harayna Institute of Technology & Management, Kurukshetra(Haryana), India. He obtained his Ph.D. in Computer Sc. and MCA From Kurukshetra University, Kurukshetra (India), M.Tech in CSE from KSOU, Karnatka (India). His Research area includes: Software Engg., Mobile Ad-Hoc Networks, Web Technology, Data Mining and Knowledge Management.

Ritika is pursuing her *Masters in Technology (CSE)* from Kurukshetra University, batch 2010-12. She completed her *Bachelors in Engineering (IT)*, batch 2007-10 from Maharishi Dayanand University, Rohtak. Prior to this, she did *Diploma in Information Technology*, batch 2004-07 from State Board of Technical Education (Haryana). She holds great academic records throughout her career. She has keen interest in *COMPUTERS SECURITY*. She has presented some papers in international journals on computer security.