ISSN:0975-9646

A Novel Teiler Public Key Cryptosystem for Securing Data in Transmit and Storage

V.Keerthi [1]
Research Scholar
Department of Computer Science
Dravidian University
Kuppam

Dr. T. Anuradha[2]
Associate Professor
Department of Computer Science
Dravidian University
Kuppam

Abstract: The growing security awareness for digital data leads to spending billions of dollars which may not give assurance for it, so there will be a continuous research in developing algorithms or techniques for security. In supporting to above in this paper we have proposed a novel cryptosystem, Teiler public key cryptosystem which uses Teiler divisors in computing keys for public key and private key which has given better performance in all stages and compared with RSA. The proposed cryptosystem can be used for securing data against attacks due to low public and private key exponents.

Keywords: **Security, Public key cryptosystem, Teiler numbers, Prime divisors, key generation, encryption, decryption.**

1. INTRODUCTION

The rise in usage of computers and communications systems in the 1960s brought with it a demand from the private sector for means to protect information in digital form and to provide security services. Providing security to a standalone computer is simple is restricted to one direction such as physical security or restricting its usage, but if we consider a computer connected to communication world the real problem arises. The security is essential to provide confidentiality, authentication, nonrepudiation, integrity which are its main goals. The science of security come under the study of cryptography which is developing day by day to address these issues. In cryptography major contributions came from late 1976 onwards by Diffie and Hellman when they published the concept of public-key cryptography and later in 1978 when Rivest, Shamir and Adleman discovered RSA, which is based on factorization problem. From then onwards the search for new public-key schemes, improvements to existing cryptographic mechanisms, and proofs of security continues at a rapid pace. Various standards and infrastructures involving cryptography are being put in place. Security products are being developed to address the security needs of an information intensive society regularly which is one of our effort in this paper [1][2].

Security is essential in different forms for information according to situation and requirement, due to the attacks that occur in different manner. The security must ensure regardless of who is involved, to one degree or another, to all parties that a transaction must have confidence and that certain objectives associated with information security have been met.The above considerations should be considered while designing an algorithm to avoid a security flaw which can always be described by an attack scenario in which some security services that the protocol purports to provide can be sabotaged by an attacker or by a number of them via their collusion. The formulated algorithm should be mathematically strong enough and should be implemented by using a robust programing language[3]. So in this paper we have developed a variant of RSA known as “Teiler RSA” which uses Teiler divisors. As the main theme of this paper is to develop an cryptosystem, we will briefly introduce the concepts of cryptography and types of cryptography in section-1, the contributions to RSA in Section-2, design of proposed Teiler-RSA in section 3 and its results in section-4 and section-5.

2. LITERATURE SURVEY

The threats to data security, from malicious attacks, viruses, spyware, and phishing to bullying, scams, and identity theft, create a feeling of vulnerability. Some of these threats are aimed directly at individuals, whereas many more target organizations with which those individuals do business. To encounter these threats and provide greater confidence, there should be a sound security techniques which are resilient to above mentioned attacks and some of them are discussed in sections below.

The first revolution event in the era of the public key cryptography was in 1976 when Diffie-Hellman [4] published their well-known paper entitled “New direction in cryptography”. This paper has suggested a great concept for public key cryptography and to build a scheme without a secure communication, but able to provide a secret communication. However, Diffie-Hellman suggested such technique for distributing the private key to be employed in the classical schemes in insecure communication channel [5]. In 1978 Rivest, Shamir and Adleman (RSA) [6] introduced the first applied scheme which is the most popular public key scheme. In 1985, Elgamal proposed a public key cryptosystem and digital signature scheme based on discrete logarithms. As to our best knowledge, this cryptosystem is still secure under discrete logarithm. But if any key k is used twice in the signing, then the system of equations is uniquely determined and secret message x can be recovered.

In 1978, RSA [6] developed a public key cryptosystem that is based on the difficulty of integer factoring. The RSA public key encryption scheme is the first example of a provably secure public key encryption scheme against chosen massage attacks. Assuming that the factoring problem is computationally intractable and it is hard to find the prime factors of n =p*q. In 1985, Elgamal [7] proposed a public key cryptosystem and digital signature scheme based on discrete logarithms which is efficient as discrete logarithm is NP hard, but it has drawbacks such as low-modulus attacks, known as plaintext attacks.

Hung-Min Sun et.al, in their work regarding RSA [8], revealed that, in RSA, the main disadvantages are, computing an exponentiation modulo N is very costly because the RSA modulus is much larger than other moduli of public key cryptosystems such as those based on elliptic curves, the size of the key pairs. In order to overcome these drawbacks, many researchers have studied variants of RSA which either reduce the computational costs [9], [10] or reduce the (key) storage requirements.

3. CRYPTOGRAPHY

Cryptography is the science of using mathematics to encrypt and decrypt concealed code and is an age-old art. some proficient argue that cryptography come out spontaneously sometime after writing was excogitate, with applications ranging from diplomatic letters to war-time disputation plans. In data and telecommunications, Cryptography enables us storage of sensitive information or transmit it across insecure networks (like the Internet) so that it cannot be read by anyone except the intended recipient, within the context of any application-to-application communication [3].

3.1 Categories of Cryptography

In olden days cryptography, called as symmetric key cryptography, involved only one key used by both sender and receiver for transforming the information. So single key must be maintained in secret was problem unable to solve. Next emerged public key cryptography involving two separate keys, one is public key and other private key linked by mathematical relations. The above two kinds of techniques are formulated as symmetric cryptography, asymmetric cryptographic techniques.

3.1.1 Symmetric Key Cryptography

In symmetric cryptography only one key is used for encryption and decryption. In symmetric-key (traditional) cryptography, both of the sender and receiver know and utilize the same secret key. The main challenge is how the key is shared secretly between the sender and receiver. If they are in another physical positions, they must hope a courier, a phone system, or some other transmission medium to check the disclosure of the secret key. Anyone who hears or tap the key in transit can later read, modify, and forge all messages encrypted or authenticated using that key. So a secure key management technique should be used. Some of the currently used cryptographic technologies in symmetric key cryptography are DES (Data Encryption Standard), IDEA (International Data Encryption Algorithm), Blowfish, RC5 (Rivest Cipher 5), AES (Advanced Encryption Standard) etc[3].

3.1.2 Asymmetric Key Cryptography

To work out the key management problem, Whitfield Diffie and Martin Hellman brought the concept of public-key cryptography (asymmetric). In asymmetric algorithm distinct keys are used to encrypt and decrypt the data. Cryptographic system needs two separate keys, one of which is secret and one of which is public. The two parts of the key pair are mathematically linked. (the ones being the integer factorization and discrete logarithm problems).while it is easy for the recipient to generate the public and private keys, to decrypt the message using the private key, and easy for the sender to encrypt the message using the public key, it is extremely difficult for anyone to derive the private key, based only on their knowledge of the public key. The most prominent Asymmetric key cryptosystem is RSA which is used even now with large key size [1].

3.2 RSA Public Key Cryptosystem Review

The RSA cryptosystem, named after its inventors R. Rivest, A. Shamir, and L. Adleman, is the most widely used public-key cryptosystem. It may be used to provide both secrecy and digital signatures. The RSA encryption method exhibits a specific feature that even an encryption key is publicly revealed it is difficult to trace the corresponding decryption key. A message is encrypted by representing it as a number M, raising M to a publicly specified power e, and then taking the remainder when the result is divided by the publicly specified product, n, of two large secret prime numbers p and q. Decryption is similar; only a different, secret, power d is used, where $e .d \equiv 1 \pmod{(p -1) . (q -1)}$. The security of the system rests in part on the difficulty of factoring the published divisor n [1][3].

3.2.1 RSA Key Generation Algorithm

Entities which need to communicate with each other; each entity creates an RSA public key and a corresponding private key. Each entity A should do the following:

1.Generate two large random (and distinct) primes p and q, each roughly the same size.

2. Compute $n = p*q$ and $\varphi(n) = (p -1)(q -1)$, where $\varphi(n)$ is the euler totient function.

3. Select a random integer e, $1 < e < \varphi(n)$, such that $\gcd(e, \varphi(n)) = 1$.

4. Compute the unique integer d, $1 < d < \varphi$, such that $ed \equiv 1 \pmod{\varphi(n)}$.

5. A’s public key is (n, e); A’s private key is d.

3.2.2 RSA Encryption

If an entity A has to send the message confidentially to an entity B, entity A should perform the following

(a) Obtain authentic public key (n, e).

(b) Represent the message as an integer M in the interval [0, n −1].

(c) Compute $C= M^e \bmod n$

(d) Send the cipher text C to A.

3.2.3 RSA Decryption

To recover plaintext M from C, A should do the following:

(a) Use the private key d to recover $M = C^d \bmod n$.

3.3 RSA Vulnerabilities

From the days of its invention, RSA is deployed in many commercial systems. It is used by web servers and browsers to secure web traffic, it is used to ensure privacy and authenticity of Email, it is used to secure remote login sessions, and it is at the heart of electronic credit-card payment systems. In short, RSA is frequently used in applications where security of digital data is a concern. The past twenty years of research [11] has led us to know about number of fascinating attacks, mostly illustrate the dangers of improper use of RSA.

3.3.1 Low Public Key and Private Key Exponent

Low Private Key Exponent

To reduce decryption time (or signature-generation time), one may wish to use a small value of d rather than a random d. Since modular exponentiation takes time linear in $\log_2 d$ (check step 4 in key generation phase of RSA above), a small d can improve performance by at least a factor of 10 (for a 1024 bit modulus). Unfortunately, a clever attack due to M. Wiener [11] shows that a small d results in a total break of the cryptosystem. Since typically' n' is 1024 bits, it follows that d must be at least 256 bits long in order to avoid this attack. This is unfortunate for low-power devices such as "smartcards", where a small d would result in big savings in computation time which result in breaking of key and become vulnerable to attacks.

3.3.2 Low Public Key Exponent

A small public exponent *e* is used to reduce the computation time of encryption or the verification time of signature. The least value for '*e*' is possibly 3 since it is a prime, but to withstand certain attacks the recommended value of *e* is 65537 ($2^{16} + 1$). When $2^{16}+1$ is used, signature verification requires 17 multiplications, as opposed to roughly 1000 when a random e $\leq \varphi$ (N) is used. Unlike the attack of the private key exponent, attacks that apply when a small e is used are far from a total break.

The most powerful attacks on low public exponent RSA are based on a theorem due to Coppersmith Theorem[11].

4. Design Of Proposed Teiler-Rsa Public Key Cryptosystem

In this research work, a Teiler cryptosystem to perform the encryption and decryption for a given message in very short span of time is proposed and to avoid the security attacks that affect the RSA algorithm reliability for secure communication. In the proposed cryptosystem, 'n' is generated from group of randomly selected divisors (known as Teiler divisors) to generate different public and private key pairs. Furthermore the suggested asymmetric cryptosystem , σ_k (n) is the sum of the k^{th} powers of the positive divisors of n, including 1 and n, where k $\in Z^+$. $\sigma_1(n)$, the sum of the (positive) divisors of n, is usually denoted by $\sigma(n)$. Since a positive number to the zero power is one, $\sigma_0(n)$ is therefore the number of (positive) divisors of n; it is usually denoted by d(n) or $\tau(n)$ (for the German Teiler = divisors)[12].

$$\sigma_k(n) = \prod_{i=1}^{\omega(n)} \frac{p_i^{(a_i+1)k} - 1}{p_i^k - 1} = \prod_{i=1}^{\omega(n)} \left(1 + p_i^k + p_i^{2k} + \cdots + p_i^{a_i k}\right).$$

Setting *k* = 0 in the second product gives

$$\tau(n) = d(n) = (1 + a_1)(1 + a_2) \cdots (1 + a_{\omega(n)}).$$

Prime divisors of 'n' have been chosen in the proposed algorithm. Further, the product of four prime divisors 'd(n)' is calculated and an integer 'u' is chosen such that 1 < u < D(n),D is prime divisor. A private key x= $(u^{d(n)})^e$ mod (n) is calculated. Security is assured with the proposed algorithm since the private key 'x' cannot be calculated using the prime divisors of 'n' which are private. Hence, the private key 'x' cannot be identified and the proposed algorithm is safe and secure.

The new Teiler Public Key Cryptosystem has three main steps;

- ✓ Key generation
- ✓ Encryption
- ✓ Decryption

4.1 Teiler RSA Key Generation Algorithm

In the key generation of Teiler Public Key Cryptosystem;

1: Select a Random integer 'n'.
2: Generate prime divisors D_1^{a1}, D_2^{a2}, D_3^{a3}…D_n^{an} of n
3: Calculate d(n) = (1+a1)(1+a2)(1+a3)……… (1+an)
4: Select a Random integer u such that 1 < u < D (n)
5: Calculate Euler function Eu = $(D_1^{a1}$-1)*$(D_2^{a2}$-1)*$(D_3^{a3}$-1)*......*$(D_n^{an}$-1)
5: Compute e value such that gcd (e, Eu) = 1 and 1< e < Eu.
4: Compute d value such that d = e^{-1} mod (Eu).
5: calculate x= $(u^{d(n)})^e$ mod (n);
6: Publish keys as Public key is (e, n) and private key is (x, d, n)

4.2 Teiler-RSA Encryption

Plain text 'M' is converted into Cipher text 'C'.

1. Obtain the recipient public key (e, n)
2. Encrypt the message m with the public key (e, n) like C = m^e mod (n) + $u^{d(n)}$.
3. Sends the cipher text C to recipient.

4.3 Teiler-RSA Decryption

Cipher Text 'C' is converted into Plain Text 'M'.

1. Uses the private key (x, d, n) to decrypt the message like
2. Before going to decryption process, the x will be decrypted as y = x^d mod (n)
3. Now finally decrypt the Cipher text as
   M= C^dmod (n) – y

4.4 Implementation Examples of Teiler-RSA Cryptosystem

Example 1:

32 Bits:

Public Key(e) = 263 Private Key(d) = 47359111743804038327
d(bits) = 66 n = 3715538693 n(bits) = 32
Input Text : SECURITY
Cipher Text :
3BA1867AD819FF58503D197283A9CD0DBC7ACEA71BEC81CDBAF22B6C06730BD
Output Text : SECURITY

Example-2

<u>64 Bits:</u>

Public Key(e) = 56263 Private Key(d) =
18182741908629691172384656489289689608371111
d(bits) = 141
n = 9999548693044318363
n(bits) = 64
Input Text : SECURITY
Cipher Text :
7DF517D5F1D0B5C1529A3D4A4D403F33125B7A16A4
666E133A57B0F7C2768CED265071798EA9DA6C32EA7
934D7DDBBD414470C423C55EF4812BA0EB0E4D9C08
B
Output Text : SECURITY

4.5 Teiler Public Key Cryptosystem Performance Analysis in contrast to RSA

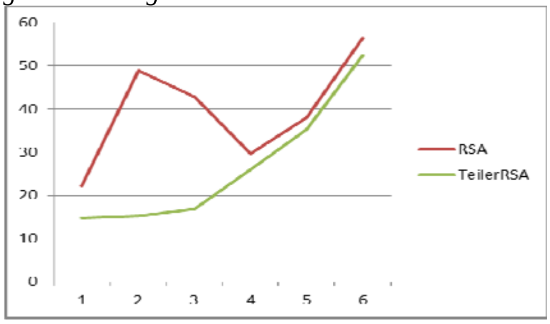
Cryptographic mechanisms provide data confidentiality by applying encryption technique to protect files against unauthorized access. Since encryption is an expensive operation, there is a trade-off between performance and security that a system designer must take into consideration. So in this paper for each phase of our Crypto System we analyzed by taking different p, q values. We have considered 32 to 1024 bit N values , analyzed and compared with traditional RSA System. Graphical analysis is given for all the three phases like Key Generation, Encryption phase and Decryption phase for the sample data.

4.5.1 Key Generation

In Teiler-RSA cryptosystem Key generation phase is critical to randomly select prime divisors to find N value, also this algorithm will use randomly selected u value to compute private key x= $(u^{d(n)})^e$ mod (n). In some cases the proposed algorithm takes more time, efficiency is compromised of some milli seconds at the cost of security which is the main aim of the system.

Time Cost for Key Generation(in milliseconds)		
N	RSA	Teiler-RSA
32	22.2	14.7
64	48.8	15.3
128	42.8	16.8
256	29.6	26
512	38.06	35.23
1024	56.35	52.34

Table -:1 Comparison of RSA and Teiler-RSA cryptosystem Key generation algorithms



Time taken on Y Axis and No of inputs on X-Axis

Figure: 1 Graph plotted showing performance of key generation in two Cryptosystems

4.5.2 Encryption

In Encryption phase the proposed algorithm has given better performance when compared to RSA and also Teiler-RSA method withstands attacks to which RSA is vulnerable. so our method generates large cipher text which is difficult to break.

Time Cost for Encryption(in milliseconds)		
N	RSA	TeilerRSA
32	0.157	0.197
64	0.081	0.256
128	0.145	0.412
256	1.254	0.812
512	4.356	5.364
1024	15.36	14.356

Table: 2 Comparison of RSA and Teiler-RSA cryptosystem Encryption process

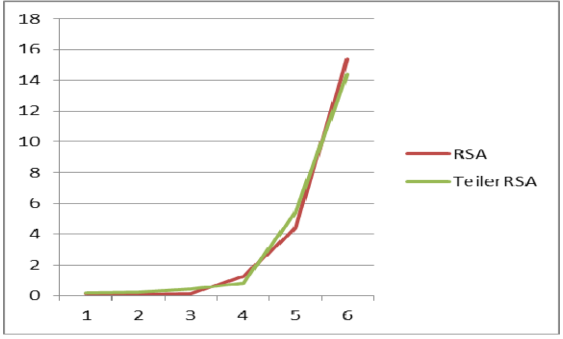


Figure 2: Graph plotted showing performance of Encryption in two Cryptosystems

4.5.3 Decryption

In Decryption phase the proposed method cipher text is complex to decrypt which is essential for high security which is shown in performance when compared to RSA for N value take up to 1024 bits.

Time Cost for Decryption(in milliseconds)		
N	RSA	Teiler-RSA
32	0.23	0.41
64	0.107	0.715
128	0.226	1.97
256	0.396	4.52
512	3.43	12.6
1024	36.152	42.36

Table : 3 Comparison of RSA and Teiler-RSA cryptosystem Decryption process

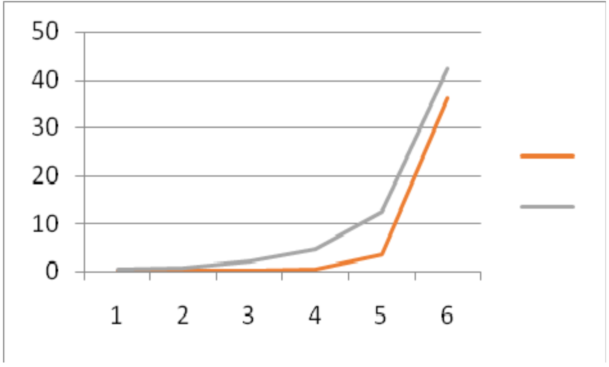


Figure: 3 Graph plotted showing performance of Decryption in two Cryptosystems

4.5.4 Algorithm Analysis in all the Three Phases

The overall time for the implementation (i.e. Key Generation, Encryption, and Decryption) of proposed cryptosystem is relatively lesser than the implementation of RSA cryptosystem.

Total Time (in milliseconds)		
N	RSA	TEILER-RSA-RSA
32	35.3	16.477
64	82	17.991
128	122	19.907
256	156	32.283
512	221	54.179
1024	440	113.876

Table .4 Comparison of RSA and Teiler-RSA cryptosystem for Key generation, Encryption, Decryption process

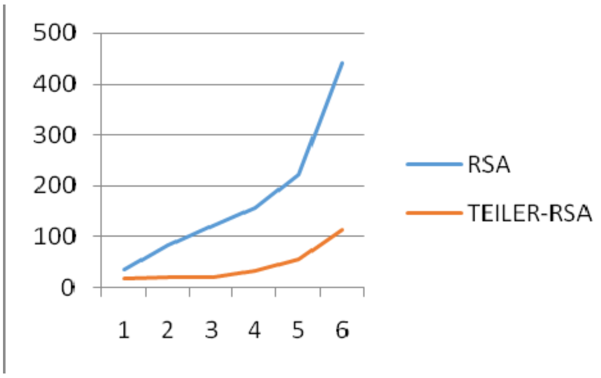


Figure: 4 Graph plotted showing performance ***(Key Generation, Encryption, Decryption)*** *in two Cryptosystems*

4.6 Teiler-RSA Public Key Cryptosystem Security Analysis in contrast to RSA

Low Private and Public Exponents:

In RSA if we use low Public key it decreases encryption time by ten times and also the speed of decryption will also be decreases if low private key exponent is chosen [15]. If we choose low public key it is to find **d** private key which leads to total breakdown of cryptosystem. But in our algorithm since we are choosing a large value 'n' and generating prime divisors D_1^{a1}, D_2^{a2}, D_3^{a3}…..D_n^{an} of 'n' and therefore it is difficult to find D_1^{a1}, D_2^{a2}, D_3^{a3}…..D_n^{an} to get **d.** In the proposed algorithm, the bit size of d is double than the bits of 'n', hence it is very difficult to break the cryptosystem.

4.7 Applications of Teiler-RSA Public Key Cryptosystem

Teiler-RSA public Cryptosystem is used for following purposes like

a) Encryption/decryption for exchange of data insecure communication channels
b) Digital signature- to digitally sign a document and for verification
c) Key exchange- in exchange of keys and secret documents securely

Teiler-RSA Public-key cryptography finds its strongest application when parties who have no prior relationship (and therefore no opportunity to establish shared secret keys) want to exchange sensitive data with each other.

The applications of online purchasing had exactly the characteristics that required public-key cryptography which can use Teiler-RSA cryptosystem to secure online merchants sensitive data like credit card numbers. Teiler-RSA cryptosystem can be used in applications for exchanging messages securely between Officials and farmers in agriculture online services, Financial service applications etc.

5.Conclusion

In this paper a new Teiler public key cryptosystem has been proposed to provide confidentiality, data integrity, non-repudiation, and authentication for data by encrypting it and decrypting it when necessary by keys. The proposed algorithm is tested with sample data and results are given along with graphical analysis. Even the latest technology like cloud computing can use Teiler cryptosystem for authenticating user to access Virtual Machine in Cloud, in identify malware process by Intrusion and detection system in Virtual machine of cloud, securing data at storage by encrypting it and in exchange of data between clients.

References

[1] Wenbo Mao, *Modern Cryptography: Theory and Practice,* Prentice Hall,2005

[2] William Stallings,*Cryptography and Network Security Principles and Practices*, 4thEdition,Prentice Hall,2005

[3]. Alfred J. Menezes et.al, *Handbook of Applied cryptography*,

[4]. Diffie W and Hellman M, "*New Direction in Cryptography*", IEEE Transaction on Information Theory, IT-22(6): 644-654, 1976

[5]. Bruce S, "*Applied Cryptography*", 2nd edition, John Wiley and Sons, Inc. 1996

[6]. Rivest R, Shamir A, and Aldeman L, "*A Method for Obtaining Digital Signatures and Public-key Cryptosystems*", Commun. ACM, vol. 21, no.2, pp. 120–126, 1978.

[7]. Elgamal T, "*A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithm*", IEEE Trans. on Inf. Theory, IT-31, No.4, pp.469-472, 1985.

[8]. Hung-Min Sun, Mu-En Wu, Wei-Chi Ting, and M. Jason Hinek, "*Dual RSA and Its Security Analysis*", IEEE Transactions on Information Theory, Vol. 53, No. 8, August 2007.

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

[10] Cesar Alison Monteiro Paixao, Decio Luiz Gazzoni Filho,"*An efficient variant of the RSA cryptosystem*".

[11]. Dan Boneh, "*Twenty Years of Attacks on the RSA Cryptosystem*", Notices of the AMS, Vol. 46, No. 2, Feb. 1999.

[12] .[Available at] https://en.wikipedia.org/wiki/Arithmetic_function