Research article Available online www.ijsrr.org ISSN: 2279–0543

International Journal of Scientific Research and Reviews

Design and Implementation of an Improved Public-Key Cryptosystem for Digital Signature

Arora Himanshu[1]* and Shrivastava Sumit[2]

[1]Computer Science & Engineering, Sunrise University, Alwar (Rajasthan), India
[2]Computer Science & Engineering, Manipal University, Jaipur (Rajasthan), India

ABSTRACT

Over the past few decades, digital communication has gain augmented interest to share information without any distance barrier. With new technology, both legal and illegal processes evolve. Therefore the applications of cyber world needing high level of safeguard for expensive data and produce explosive growth to the field of data security, process of protecting information, its availability, privacy and integrity. However, in recent years, a lot of research has taken place in direction to trim down the security issues by contributing various approaches but different terrains pose separate challenges. In this context to fill the gap of security issues, this paper has present a new approach to improve the performance of accessible cryptography algorithm, namely RSA (Rivest, Shamir and Adleman) algorithm. The proposed approach has claimed to be more efficient than the already existed algorithms.

KEYWORDS: Asymmetric Cryptography, Dual Modulus, Message digest, Random number, Discrete logarithm, Factorization

***Corresponding Author:**

Himanshu Arora

Research Scholar, Computer Science & Engineering,

Sunrise University, Alwar (Rajasthan), India

E Mail - arora_himansh@yahoo.com

INTRODUCTION

In the present era, information is wealth of any organization and everyone wants the secrecy and safety of their confidential data. However, a lot of methods have been widely used since long past to achieve security goal but different trains pose separate challenges. Most of the proposed methods use the idea of cryptography.

Cryptography is the science of secret writing, converting messages or data into a different form to exchange messages between two parties who want the communication over an insecure channel. Without the right knowledge of the key no-one can access the correct information [1, 2]. The cryptography can be further divided in terms of symmetric and Asymmetric key cryptography. Where symmetric key cryptography technique use a single key to encrypt and/or decrypt the secrete data, Asymmetric key cryptography has use two different keys for the same purpose. One of the key has used for the encryption and for the decryption second key is used by the receiver. The beauty of this scheme is that every communicating party needs just a key pair for communicating with any number of other communicating parties. Once someone obtains a key pair, he /she can communicate with anyone else. Asymmetric key cryptography is also known as public key cryptography.

A digital signature (scheme) is a mathematical scheme for demonstrating the authenticity of a digital message or document. Digital signatures are commonly used for software distribution, financial transactions, and in other critical security areas where it is important to safeguard against forgery and tampering [2,4].

RSA (named after Rivest, Shamir and Adleman who first publicly described it) is an algorithm for asymmetric cryptography. It is the first algorithm known to be suitable for signing as well as encryption, and was one of the first great advances in public key cryptography. RSA is widely believed to be secure given sufficiently long keys.[4]

The security of the RSA algorithm lies in the fact that there is no good way of factorization of large prime numbers. As long as no one finds a way to solve this problem in reasonable time, RSA will be safe and secure encryption algorithms.

Due to advancement in technology and improvement in computation speed it may become possible to break the RSA, so a new technology is essential in times to come [5].

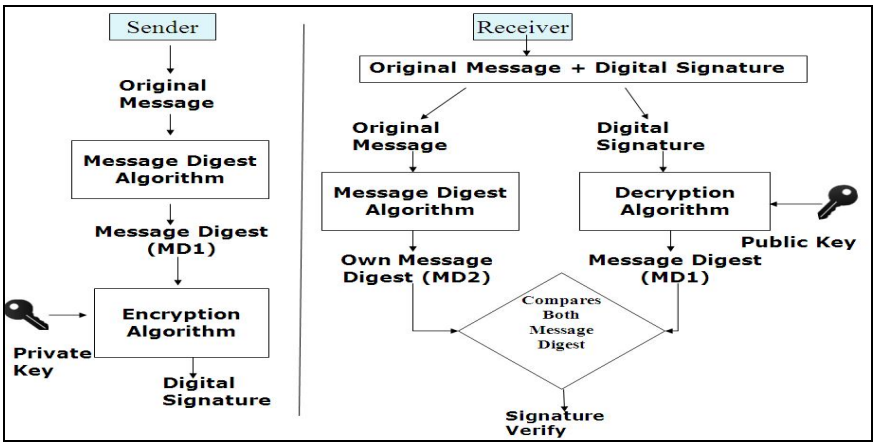

Fig. 1 Mechanism of the Digital Signature

In this context, to improve the security of secrete message this paper has present a novel approach by dual modulus with discrete logarithm based technique

HITTING METHODOLOGIES TO RSA

There are four possible methodologies to attack the RSA algorithm[6].

Brute Force:

An attacker can use this type of attack technique against any encrypted data to look for all possible private keys. It comprises of methodically examining all the possible keys in anticipation of the exact key to be found. In the worst situation, it would include traversing through the whole search space [7, 8].

Mathematical Attacks:

The following methods can be used to hit the RSA algorithm mathematically.

i. Isolating k into its two prime aspects. It would make easy inference of φ (k) = (p – 1) x (q – 1) which would successively agree to to determine d ≡ e-1 (mod φ (k)).

ii. Finding out φ (k) straightforwardly devoid of discovering p and q. Yet again, this would permit the resolving of d ≡ e-1 (mod φ (k)).

iii. Resolve d directly without finding out φ (k) first.

iv. Determining d given e and n seems to be as time-consuming as the factoring problem by means of current well-known algorithms. So, factoring performance can be used as a point of reference for evaluating the security of RSA[7, 8].

Timing Attacks:

The timing attack is a cipher text-only attack that can be modified to operate with any implementation which does not run in fixed time, basically a way of deciphering a user's private key information by measuring the time it takes to carry out cryptographic operations[5]. In this algorithm, modular exponentiation is carried out bit by bit, by one modular multiplication to execute iteration every time and an additional modular multiplication performed for each 1 bit[7, 8].

Chosen Cipher Text Attack:

The fundamental RSA algorithm is exposed to a chosen cipher-text attack (CCA). CCA is described as an attack in which opponent picks up a number of cipher texts and is then given the equivalent plain texts which are decrypted with the target's private key. Hence, the adversary could choose a plain text, encrypt it by means of the target's public key and then be able to obtain the plain text back by having it decrypted by means of the private key[7, 8].

PROPOSED WORK

To improve the security of RSA, a new Digital Signature algorithm is proposed, presented and named Asymmetric Digital Signature Algorithm based on Dual Modulus and Discrete Logarithm Concept (ADSDMDL). By using an efficient implementation of ADSDMDL algorithm, performance of the algorithm is analyzed by changing various parameters of the algorithm.

This algorithm has the following features

- Double modulus based encryption using two Private keys
- Double modulus based decryption using two Public Keys
- More than two large prime numbers used in generating modulus
- Three Natural numbers used to increase the security based on discrete logarithm.

The features cited above increase the security of message but the time for digital signature creation, digital signature verification and key generation is also increased. As the security of RSA/ADSDMDL cryptosystem is based on both the factorization of modulus number and concept of discrete logarithm. So in ADSDMDL cryptosystem, double modulus scheme is used with two large public and private keys hence it is more difficult factoring of two modulus says n_1& n_2 and getting the private key. In addition, it

uses three natural numbers. These natural numbers increase the security of the cryptosystem because of discrete logarithm nature. Although the above changes degrade performance of the algorithm in terms of key generation time, digital signature creation time and digital signature verification time, the security is exponentially improved & one has to solve both the problems to break ADSDMDL algorithm.

COMPARISON BETWEEN RSA AND ADSDMDL CRYPTOSYSTEM

In ADSDMDL cryptosystem dual modulus scheme is used, it's is quite complicated to factor dual modulus as compared to single modulus and the performance of the ADSDMDL algorithm slow down as compared to general RSA but the security is exponentially increased. Following Table 1 shows the comparison between RSA cryptosystem and ADSDMDL cryptosystem.

Table 1: comparison between RSA cryptosystem and ADSDMDL cryptosystem

RSA algorithm	**ADSDMDL algorithm**
Single modulus scheme is used	Double Modulus scheme is used
Two different keys (one for signing and another for verification) are used.	Two different key Pairs (one key pair for signing and another for verification) are used.
Digital Signature Creation and Verification time is less.	Digital Signature Creation and Verification time is near about double
Single modulus scheme is used so the security is less.	Security is exponentially increased as one attacker have to factor double.
It used for encryption & decryption.	It also used encryption & decryption.
Hardware time and software time complexity is $O(k^2)$ and $O(k^3)$	Hardware time and software time complexity is double as compare to RSA but still complexity is $O(k^2)$ and $O(k^3)$
Applicable in multi user environment.	It is also applicable in multi user environment.

Other face of ADSDMDL algorithm is that it's Key generation process is slow then RSA algorithm as it calculates all operations twice such as generating large prime numbers, taking modulus, generating public and private keys. But key generation time has no importance in Public key cryptosystem as it is only once when keys are generated, afterward only signing and verification time matters. By analyzing the simulation results, it is clear that overall performance of ADSDMDL is better in terms of security despite of slower speed.

Simulation Results

For the simulation purpose of the purposed cryptosystem involves working with large integers (i.e. 1024 bits). There are several libraries to consider for application herein such as the BigInteger library (Java) [9], the GNU MP Arbitrary Precision library (C/C++), and the OpenSSL crypto library (C/C++). As the application is developed in JAVA, the BigInteger library is used. BigInteger library provides operations for modular arithmetic, GCD calculation, primarily testing, prime generation, bit manipulation, and other miscellaneous operations. Evaluation time is a machine dependent task which is required to be implemented on a particular system. Once the system configuration is changed, evaluation time will also be changed accordingly, however, in this work, following system configuration is used [10, 11].

- Operating System: Windows XP Professional (5.1, Build 2600) Service Pack 2
- Processor: Intel Pentium Dual CPU E2200 @ 2.20GHz (2 CPUs)
- Memory: 1024MB RAM

Changing the prime number (p & q) Size:

The value of prime numbers p and q will affect the other parameters as shown in table that when the value of prime numbers is increased, the overall execution time which comprises digital signature creation time and digital signature verification time gets enhanced rapidly. When the size of prime umber is increased from 128 to 256 the total execution time taken is less in proportion to when the average bit size is increased from 512 to 1024 bits.

Table 2: Effect of changing the size of prime on digital signature creation time and digital signature verification time, taking size of Public key 128 bits, size of chunk 128 bits.

Prime number (p_1,p_2 & q_1,q_2) size (bit)	**ADSDMDL cryptosystem**			**RSA Cryptosystem**		
	Digital Signature Creation Time(A) (ms)	**Digital Signature Verification Time(B) (ms)**	**Total Execution Time(A+B)**	**Digital Signature Creation Time(A) (ms)**	**Digital Signature Verification Time(B) (ms)**	**Total Execution Time(A+B)**
128	281	219	500	156	78	234
256	1437	437	1874	719	266	985
512	9641	1594	11235	4812	875	5687
1024	71469	5860	77329	35531	3235	38766

Changing the Public Key Length:

Changing the maximum limit of the length of the public key (e-bit) affects the key generation time as well as signature verification time of both the algorithms but it does not affect the signature creation time as there is no role of public key.

Table 3: Effect of changing the Public key size on digital signature creation time and digital signature verification time, taking size of prime number 1024 bit and size of chunk 1024 bit.

Public Key Size (bit)	ADSDMDL cryptosystem			RSA Cryptosystem		
	Digital Signature Creation Time (A) (ms)	Digital Signature Verification Time(B) (ms)	Total Execution Time(A+B)	Digital Signature Creation Time (A) (ms)	Digital Signature Verification Time(B) (ms)	Total Execution Time(A+B)
128	35157	1281	36438	17500	640	18140
512	34953	4672	39625	17500	2391	19891
1024	35344	9188	44532	17562	4562	22124

Comparison by changing the length of the message to be processed (Chunk Size)

The chunk size is the number of characters to be processed at a time, either in digital signature creation or signature verification process. Here the message is divided into sub blocks each of length equal to chunk size. To illustrate the importance of this parameter, the message is taken long enough and the chunk size is allowed to vary in both the signature creation/signature verification process. The table indicate that increase in chunk size is responsible for decrease in total execution time.

Table 4: Effect of changing the Chunk size on digital signature creation time and digital signature verification time, taking size of prime number 1024 bit and public key size128 bit.

Chunk Size (bit)	ADSDMDL cryptosystem			RSA Cryptosystem		
	Digital Signature Creation Time (A) (ms)	Digital Signature Verification Time(B) (ms)	Total Execution Time(A+B)	Digital Signature Creation Time (A) (ms)	Digital Signature Verification Time(B) (ms)	Total Execution Time(A+B)
128	71469	5860	77329	35531	3235	38766
256	35922	2688	38610	17953	1343	19296
512	18250	1313	19563	9109	672	9781
1024	9125	672	9797	4563	328	4891

CONCLUSION:

This paper presents a modified version of RSA algorithm called Asymmetric Digital Signature Algorithm based on Dual Modulus and Discrete Logarithm Concept (ADSDMDL). ADSDMDL is a double modulus based public key cryptosystem and more secure against Brute force attack as compared to RSA, To improve the security, in ADSDMDL cryptosystem dual modulus (n1 & n2) and concept of discrete logarithm are used hence the complexity of factoring the modulus is increased exponentially. It is evident from the simulation results and table that the overall performance in terms of security, ADSDMDL is better then RSA but in terms of execution time of digital signature creation, signature verification and key generation process, RSA is better. If the keys size and/or modulus size in ADSDMDL are kept half of the RSA keys and/or modulus size then a better performance in terms of security and speed can be achieved by the ADSDMDL algorithm.

REFERENCES

1 William Stallings, “Cryptography and Network Security”, ISBN 81-7758-011-6, Pearson Education, Third Edition, pages 42-62,121-144, 253-297.

2 AtulKahate, “Cryptography and Network Security”, ISBN-10:0-07-064823-9, Tata McGraw-Hill Publishing Company Limited, India, Second Edition, pages 38-62,152-165,205-240,340-370.

3 R. Rivest, A. Shamir and L. Adleman. “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”,Communications of the ACM, February 1978; 21 (2): 120-126.

4 Hong Jingxin, “A New Forward-Secure Digital Signature Scheme”, IEEE International Workshop on Anti-counterfeiting, Security, Identification, April 2007; 254-257.

5 Bryan Poe, “Factoring the RSA Algorithm”,Mat / CSC 494, April 27, 2005; 1-6.

6 Richard E. Smith “Internet Cryptography “, ISBN 81-297-0351-3, Pearson Education, pages 33-85.

7 Christpher M. king, Curtis E. Dalton, & T. Ertem Osmanolu, “Security Architecture Design, Deployment & Operations”, ISBN-0-07-047272-6, Tata McGraw-Hill Publishing Company Limited, New Delhi, pages 91-93, 347-370.

8 Alfred Menezes, “Evaluation of Security Level of Cryptography: RSA-OAEP, RSA-PSS, RSA Signature”, University of Waterloo, 2001; 4-13.

9 AllamMousa , "Sensitivity of Changing the RSA Parameters on the Complexity and Performance of the Algorithm", ISSN 1607 – 8926, Journal of Applied Science, Asian Network for Scientific Information, 2005; 60-63.

10 Li Xiao-fei. "An Improved ElGamal Digital Signature Algorithm Based on Adding a Random Number", Second International Conference on Networks Security, Wireless Communications and Trusted Computing (NSWCTC), April 2010; 2: 236-240.

11 Wen-bi Rao, Quan Gan "The Performance Analysis of Two Digital Signature Schemes Based on Secure Charging Protocol", International Conference on Wireless Communications, Networking and Mobile Computing, Sept. 2005; 2: 1180 - 1182.