# Enhancement in the security of RSA algorithm Using Subset Sum Cryptography

Ravi Kumar[#1], Praveen Kumar Verma[*2], Maneesh Pant[#3]
[#]*Department of Computer Science[1,2,3], , Shri Ram Group of Colleges[1], Roorkee Institute of Technology[2], College of Engineering Roorkee[3]*
[1]balyan.ravi@gmail.com
[2]praveenbit08@gmail.com
[3]maneeshgbpuat@gmail.com

***Abstract*—We propose a new optimized solution for the RSA (Rivest–Shamir–Adleman) algorithm with the help of subset-sum cryptosystem. We will also use the super increasing sequence for key generation process of RSA algorithm. In the RSA cryptosystem, there are problem of brute-force attack, factorization attack and mathematical attack. In proposed algorithm, we will use the hybrid combination of the subset sum problem and RSA cryptosystem. If a hacker or intruder wants to break our proposed systems then they have to factor the modulus into its primes as well as find the secret set A. If RSA which is based on single module, is broken in time x and subset sum algorithm is broken in time y then the time required to break this proposed algorithm is x*y. So the security of our proposed system is increased as compare to RSA algorithm.**

***Keywords*— Cryptography, Subset Sum, Public key, Private Key, RSA, Super – Increasing Sequence**

## I. INTRODUCTION

Cryptography is probably the most important aspect of any online digital communication security (or as a matter of fact offline also) and is becoming increasingly important as a basic building block for computer security. Symmetric encryption is a form of cryptosystem in which encryption and decryption are performed using the same key. Asymmetric encryption is a form of cryptosystem in which encryption and decryption are performed using different key i.e. public key and private key.

RSA is based on the principle that some mathematical operations are easier to do in one direction but the inverse is very difficult without some additional information. In case of RSA, the idea is that it is relatively easy to multiply but much more difficult to factor. Multiplication can be computed in polynomial time where as factoring time can grow exponentially proportional to the size of the number. RSA consist of three steps:

1. Key Generation Process
2. Encryption Process
3. Decryption Process

### Security of RSA

The security of RSA cryptosystem is also broken by two attacks based on factorization attack and chosen-cipher text attacks.

**Factorization Attack:** The security of RSA is based on the idea that the modulus is so large that it is infeasible to factor it in a reasonable time.
Bob select p and q and calculate n=p*q.
Although n is public, p and q are secret. If Eve can factor n and obtain p and q, she can calculate $\Phi = (p-1) \times (q-1)$.
Eve then calculate $d \equiv 1 \pmod{\Phi}$ because e is public. The private exponent d is the trapdoors that Eve can use to decrypt any encrypted message.

**Choose a cypher text attack:** Any positive attack on RSA technique can be based on any one of the above given attack factors.

## II. SUB-SET SUM CRYPTOGRAPHY

The subset sum problem is an important problem in complexity theory and cryptography. The problem can be summarised as follows: given a set of integers such that sum of some non-empty subset equal exactly zero. For example, given the set {−7, −3, −2, 5, 8}, the answer is yes because the subset {−3, −2, 5} sums to zero. The problem is Non Polynomial-Complete class (NP-Complete). We can divide the "sum problem" into two problems commonly known as the "subset sum problem". The first is the problem of finding what subset of integers has the required sum. This is also an integer relation problem where the relation coefficients are 0 or 1. The second subset sum problem is the problem of finding a set of 'n' distinct positive real numbers with as large collection as possible of subsets with the same sum.
The subset sum problem is a good introduction to the NP-complete class of problems. There are two reasons for this:

1. It is a decision and not an optimization problem
2. It has a very simple formal definition and problem statement.

The Subset-Sum cryptosystem (Knapsack Cryptosystem) is also an asymmetric cryptographic technique. This system is also based on the subset sum problem (a special case of the knapsack problem): An instance of the Subset Sum problem is a pair (S, t), where S = {x1 , x2 , ..., xn} is a set of positive integers and t (the target) is a positive integer. The decision problem asks for a subset of S whose sum is as large as possible, but not larger than t. However, if the set of numbers (called the knapsack) is super increasing, that is, each element of the set is greater than the sum of all the numbers before it; the problem is easy and solvable in polynomial time with a simple greedy algorithm.

## III. Related Work

The research work on cryptography dates back to 1978 when Merkel [1] used knapsack algorithm for encrypting or hiding the data. A secret key was needed to access the system designed by [1]. Elgamel [2] in 1985 designed a public key cryptosystem. He used Diffie-Hellman system for electronic key distribution. In [3] the authors tried to develop an algorithm improving the qualities of RSA. They improved on the time taken by RSA but failed to address issues of Session key attacks and replay attacks by intruder. Ramasamy [4] used a knapsack based encryption/decryption technique by which they improved on the aspect of brute force attack but their algorithm failed to establish group key authentication and also vulnerable to man in the middle attack. In [5] author used knapsack based elliptic curve cryptography method for securing the system but not able to mitigate the replay attack and phishing attack.
In the RSA cryptosystem there are problem of brute-force attack [6] factorization attack and mathematical attack. In our proposed work we have designed and implemented an algorithm which is based on the hybrid combination of the subset sum problem and RSA cryptosystem. If anyone wants to break our proposed systems then they have to factor the modulus into its primes as well as find the secret set A. If RSA, which is based on single module, is broken in time x and subset sum algorithm is broken in time y then the time required to break this proposed algorithm is x*y. This is a huge time to break this system. So the security of our proposed system is increased as compare to RSA algorithm. Also, the proposed system is able to mitigate man in the middle attack, phishing attack and replay attacks.

## IV Proposed RSA Algorithm using Subset Sum Cryptosystem

In this paper, we propose a new RSA algorithm [7] based on Subset Sum cryptosystem Knapsack. The proposed scheme is divided into three phases: Key Generation Process, Encryption Process and Decryption Process. In this section we introduce a new approach for public key cryptosystem. Modified Subset Sum (MSS) is an asymmetric-key cryptosystem in which two keys are required: a public key and a private key. Furthermore, unlike RSA, it is one-way, the public key is used only for encryption, and the private key is used only for decryption. Thus it is useless for authentication by cryptographic signing. Modified algorithm consists of following three steps:

**Step 1. Key Generation process**

1. Generate two large random primes, p and q, of approximately equal size such that their product M = p x q is of the required bit length, e.g. 1024 bits. (From Big Integer library function of Java)

2. Now calculate M = p * q and $\phi(n) = (p-1) * (q-1)$.

3. Randomly choose an integer 'e' ,which satisfying $1 < e < \phi(n)$, such that gcd $(e, \phi(n)) = 1$.

4. In this step calculate the secret exponent d, $1 < d < \phi(n)$, such that e x d 1 (mod $\phi(n)$).

5. Choose a super increasing set $A = (a_1, ..., a_n)$

6. Choose an integer M with M>SUM i=1...n($a_{i)}$. M is called modulus.

7. Choose a multiplier W such that gcd (M, W) = 1 and 1<=W<M
   This choice of W guarantees an inverse element U: UW = 1 (mod M)

8. To get the components $b_i$ of the public key B, perform $b_i = a_i$*W mod M, I = 1 ... n. The super increasing property of A is concealed by modular multiplication.

The public key is (B, n, e) and the private key is (A, M, W, n, d). Keep all the values d, p, q and $\phi(n)$ secret. Public key is published for everyone and private key must be kept secret. Then by using these keys encryption and decryption are performed.

**Step 2. Encryption of Message**

Sender A does the following:

The length of a message to be encrypted is fixed by the parameter n prior to encryption; a possibly larger message p has to be divided into n-bit groups.

Let p = (p1, p2... pn) the message to be encrypted.

i. The cipher text c is obtained by computing
   c = b1p1 + b1p2 + ... + bnpn
ii. Computes the cipher text $c_1 = c^e \bmod n$.
iii. Sends the cipher text $c_1$ to B.

**Step 3. Decryption of Message**

Recipient B does –

i. Uses private key and first compute $m_1 = c_1^d \bmod n$
ii. First compute $c' = Um_1 \bmod M = W^{-1} c \bmod M$
iii. Now solve (A, c'). Because A is super increasing , (A, c') is easily solvable.

Let $X = (x_1... x_n)$ be the resulting vector and $p_i = x_i$ and $p = (p_1... p_n)$ is the plaintext.

**Security analysis of RSA cryptosystem**

Two possible approaches to attacking the RSA algorithms are as follows:

1. Brute force: This involves trying all possible private keys.
2. Mathematical attacks*:* These are based on factoring the product of large primes such as factor n into its prime factors p and q respectively then calculating Φ, which, in turn, enables determination of d = e-1 mod Φ.

**Cryptanalysis**

Cryptology consists of two parts: cryptography and cryptanalysis. The former is to deal with the design of algorithms, protocols, and systems which are used to protect information against specific threats. The latter is used in mathematical methods to prove or check whether the design achieves a particular security goal. Also, can it withstand an attack from the list of threats given in the security specification of the design. The general definitions for each of several basic types of cryptanalytic attacks are given as follows:

1. **Cipher text-only attack**. As its name indicates, the cryptanalyst only has the cipher texts of several messages, all of which have been encrypted using the same encryption algorithm. By observing the cipher texts, the cryptanalyst tries to recover the messages as many as possible, or even to deduce the key used to encrypt the messages. In the history of cryptography, statistical techniques such as frequency analysis were developed for the cipher text-only attack. Early cipher implemented using pen-and-paper could be broken by this method.

**2. Known-plaintext attack**. The cryptanalyst not only has the cipher texts, but also has the corresponding plaintext to those cipher texts. If the cryptanalyst is able to deduce the key used to encrypt the plaintexts, the attack would be successful. Classical ciphers are typically vulnerable to known plaintexts attack. A Caesar cipher could be broken by a frequency analysis on the cipher text and corresponding plain text, and exhausted search for the small key space (there are only 26 keys/english characters in the Caesar cipher).

**3. Chosen-plaintext attack.** A chosen-plaintext attack is a model in which cryptanalyst has capability to choose arbitrary plaintexts to be encrypted and obtain the corresponding cipher text. This attack model becomes quite important in the context of public key cryptography, because the encryption keys are public. Attackers can encrypt any plaintext that they want and then obtain the cipher- texts. Generally speaking, any cipher that can prevent chosen-plaintext attacks is then also guaranteed to be secure against known-plaintext and cipher text-only attacks.

**4. Chosen-cipher text attack.** A chosen-cipher text attack is primarily applicable to asymmetric cryptographic algorithms. The cryptanalyst starts with the cipher texts to be decrypted with unknown keys, and then obtains the corresponding decrypted plaintexts. Subsequently, the cryptanalyst tries to deduce more information about the key.

## V PROPOSED ARCHITECTURE

In the proposed architecture, the subset sum problem and super increasing sequence is applied to the key generation, encryption and decryption to provide high security to RSA cryptosystem.

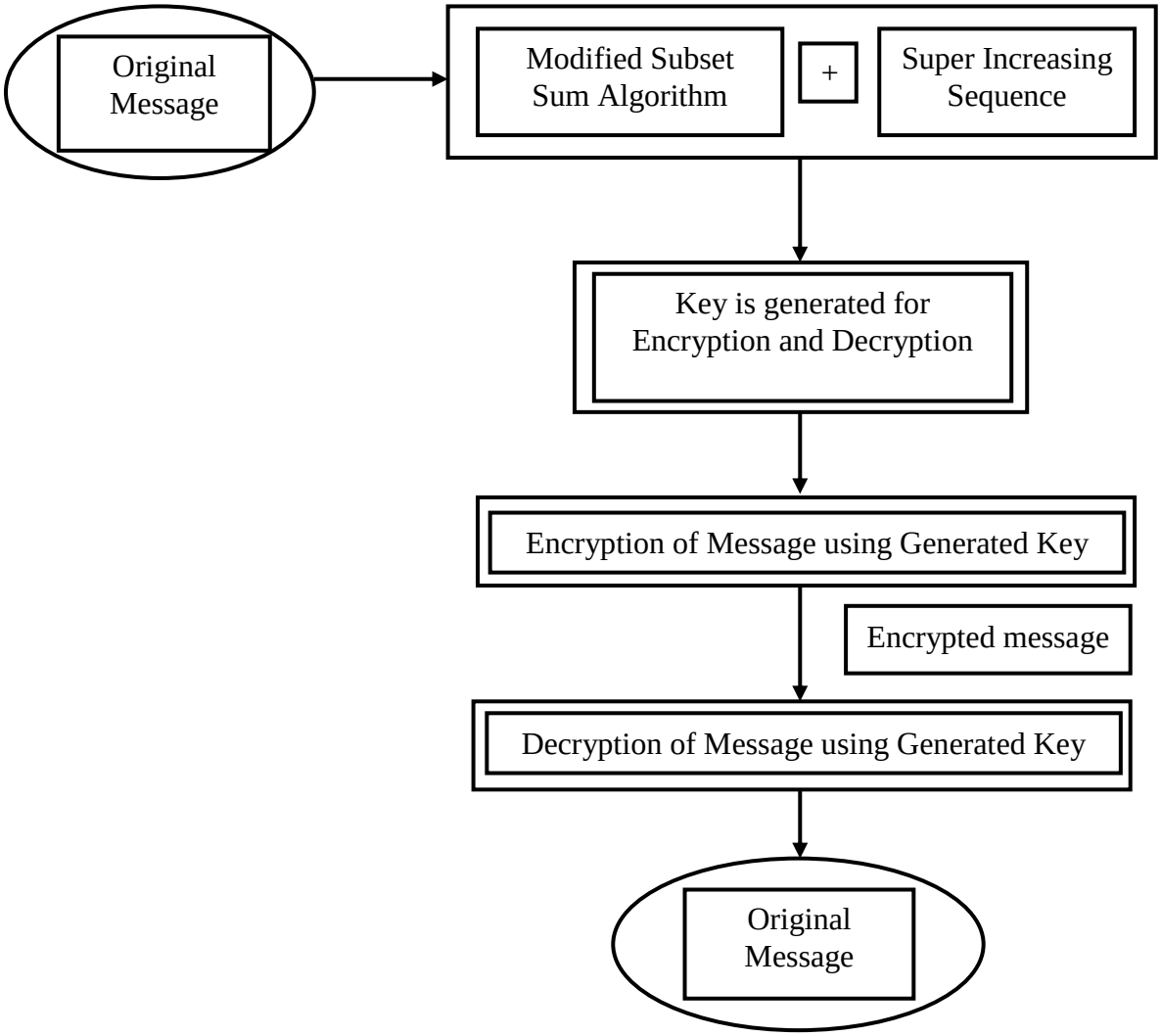


Fig 1: Block diagram of Proposed Enhanced RSA Cryptosystem

## VI. COMPLEXITY OF THE PROPOSED SYSTEM

The RSA cryptosystem is well – known asymmetric cryptosystem in which the computational complexity for both of the encryption and decryption are of the order of k3, where k is the number of bits of the modulus n. The computational complexity of subset sum problem is O(k), if a super increasing set with greedy approach is used where k is the number of items in set-A. So in this way, computational complexity of our proposed cryptosystem is order of O (k + k3) i.e. O (k ) or encryption and on the order of k for decryption.

VII. Comparison of RSA and Schmidt-Samoa cryptosystem (SSC) over RSA

**Simulation Process and Results**

For the simulation purpose, SSC is implemented as a user-friendly GUI. This GUI application is implemented using Java Big Integer library functions. In this application, one can either enter the prime numbers or can specify the bit length of the prime numbers to generate automatically. Big Integer library provides operations for modular arithmetic, GCD calculation, primarily testing, prime generation, bit manipulation, and a few other miscellaneous operations.

The simulation of the algorithm, implemented in JAVA, running on a 2 GHz P-IV Processor and 512 MB RAM, using a 1000 characters long message for encryption/decryption. The PKC algorithms (RSA & SSC) have some important parameters affecting its level of security and speed. The complexity of decomposing the modules into its factors is increased by increasing the module length. This also increases the length of private key and hence difficulty to detect the key. Another parameter is the number of items in set A. As the number of items in set A increases, the size of the message which is encrypted at a time also increases, hence the security also increases as well as difficulty of detecting the private set A from public set B also increases. The RSA and SSC parameters are changed one parameter at a time and the others are kept fixed to study the relative importance. The key generation, encryption, decryption time is depends on the speed of the processor and the RAM. Table 1 shows the simulation results of both the algorithms.

**Changing the modulus length**

Changing the modulus affects the other parameters of the algorithms as shown in Table 1. It is clear here that increasing the modulus length (bits) increases the bit length of their factors and so the difficulty of factoring them into their prime factors. Moreover, the length of the secret key ‘d’ increases at the same rate as n-bit increases. As a result, increasing the n-bit length provides more security. On other hand by increasing the n-bit length increases the values of key generation time, encryption time and decryption time. Hence increasing the n-bit length increases the security but decreases the speed of encryption, decryption and key generation process as illustrated by Figure 1 and 2.

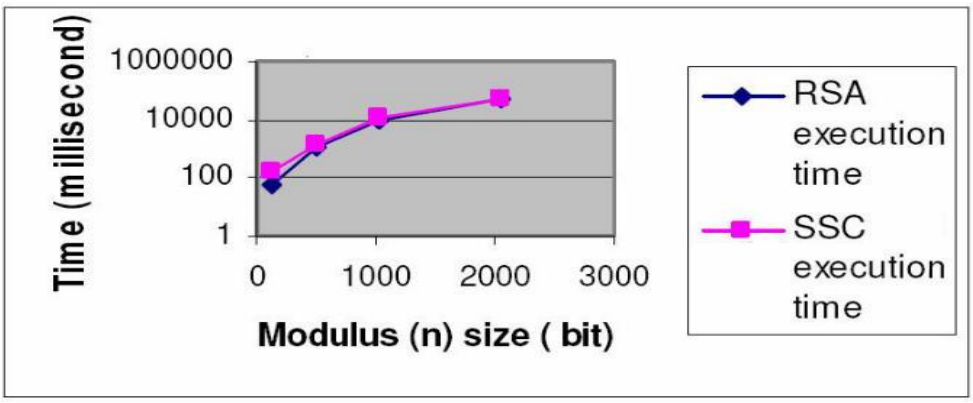


Fig 2. Modulus size v/s RSA & SSC algorithm’s execution time, taking number of items in set A 128 and size of Public Key 128 bit

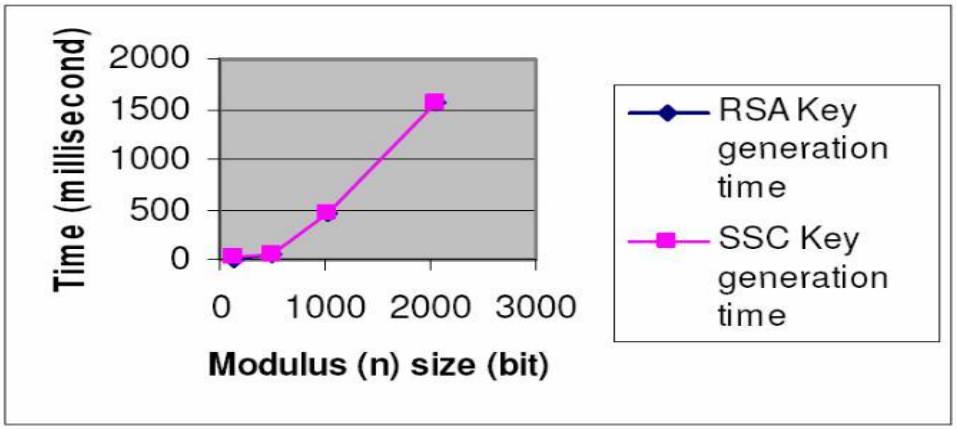


Fig 3. Modulus size v/s Key generation time, taking size of Public key 128 bit and number of items in set A are 128

TABLE I
EFFECT OF CHANGING THE MODULUS LENGTH AND SIZE OF SET A ON THE SIZE OF PRIVATE KEY, KEY GENERATION TIME, ENCRYPTION TIME AND DECRYPTION TIME, WHILE THE SIZE OF PUBLIC KEY HAS KEPT CONSTANT (128 BIT)

| Size of N (bit) | Size of d (bit) | No. of Element In Set A | SSC | | | | Proposed Enhanced RSA cryptosystem | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Key generation time (ms) | Encryption time (ms) | Decryption time(ms) | Total Execution Time(ms) | Key Generation time (ms) | Encryption time (ms) | Decryption time (ms) | Total Execution Time (ms) |
| 128 | 128 | 32 | 18 | 235 | 78 | 329 | 16 | 94 | 62 | 172 |
| 128 | 128 | 64 | 16 | 94 | 47 | 156 | 0 | 16 | 47 | 63 |
| 512 | 512 | 32 | 130 | 1203 | 1766 | 3094 | 109 | 563 | 1719 | 2391 |
| 512 | 512 | 64 | 70 | 344 | 875 | 1282 | 63 | 141 | 859 | 1063 |
| 512 | 512 | 128 | 80 | 172 | 453 | 703 | 47 | 78 | 422 | 547 |
| 1024 | 1024 | 32 | 692 | 5407 | 11328 | 17423 | 688 | 1719 | 12172 | 14579 |
| 1024 | 1024 | 64 | 462 | 6593 | 5735 | 12781 | 453 | 2968 | 5688 | 9109 |
| 1024 | 1024 | 128 | 568 | 516 | 2859 | 3937 | 515 | 219 | 3344 | 4078 |
| 1024 | 1024 | 512 | 12412 | 187 | 782 | 13780 | 281 | 47 | 735 | 1063 |

## VIII. CONCLUSION

This paper presents an extended version of Subset-Sum problem over RSA algorithm called Extension of cryptosystem through subset-sum over RSA. The modified subset sum algorithm with super increasing function gives us a secure key which is used for encrypting and decrypting the message. Our algorithm is can be used in Internet of things where the data sent and received is through wireless sensor network. Our algorithm can mitigate most of the attacks. The proposed algorithm can be modified to mitigate viruses like Trojan horse which is targeted as future work.

## ACKNOWLEDGMENT

We would like to give our sincere thanks to faculty and support staff of Department of Computer Science and Engineering, Shree Ram Group of Colleges for providing the facilities to conduct the research.

REFERENCES

**1**. R. Merkle and M. Hellman, "Hiding information and signatures in trapdoor knapsacks," in IEEE Transactions on Information Theory, vol. 24, no. 5, pp. 525-530, September 1978, doi: 10.1109/TIT.1978.1055927.

2. T. Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," in IEEE Transactions on Information Theory, vol. 31, no. 4, pp. 469-472, July 1985, doi: 10.1109/TIT.1985.1057074.

3. An Efficient Signature System Using Optimized RSA Algorithm IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.12, December 2008 Prof. MOUSTAFA ABD EL-AZIEM, Dr. MOHAMMAD ALI GOMAA.

4. Knapsack based ECC Encrytion and Decryption. International journal of network security, Vol.12,No.1,jan 2011.R.Rajaram Ramasamy and M. Amutha prabakar.

5. Digital signature scheme with message recovery using knapsack-based ECC International journal of network security, Vol.12,No.1,jan 2011.R.Rajaram Ramasamy and M. Amutha prabakar.

6. W. Stalling, *Cryptography and Network Security*5th Edition, 2006 Pearson Education, Inc., publishing as Prentice Hall, USA

7. Behrouz A. Forouzan. And D. Mukhopadhyay *Cryptography and Network Security*, 2nd edition, 2008 Tata McGraw Hill Companies, Inc., New York