# ADVANCED SECURE METHOD FOR DATA TRANSMISSION IN MANET USING RSA ALGORITHM

**Bello Musa Yakubu[1], Mr. Pankaj Chajera[2], Dr. Ahmed Baita Garko[3]**

*[1,2]Departement of Computer Science, Sharda University, (India)*

*[3] Depatment of Comuter Science, Fedral University Dutse, Nigeria*

## ABSTRACT

*In this work, a discussion is made based on a novel method for the data security authentication in mobile ad hoc network using the combination of symmetric and asymmetric algorithms.*

*In order to have data security, all the data packets are encrypted and decrypted using a private key and authentication can be obtained by asymmetric cryptography all using RSA algorithm. To ensure the proper complexity of RSA algorithm for higher level security, an enhanced RSA cryptographic algorithm is proposed.*

***Keywords: RSA algorithm, MANET, Encryption, Decryption, Trapdoor, Enhanced RSA algorithm (ERSA)***

## I. INTRODUCTION

The security in the network plays an important role and can be achieved by cryptographic algorithms [ HYPERLINK \l "Sal" 1 ]. Cryptography is the science and art of transforming messages to make them secure and immune to attacks by authenticating the sender to receiver within the network. The cryptographic algorithms are of two types 2]**}** symmetric key and asymmetric key algorithms. Symmetric key algorithm uses single key to encrypt and decrypt the data whereas, asymmetric key algorithm uses two types of keys i.e. public key for the encryption and private key for the decryption. Two important properties of crypto systems are its speed and security. Speed refers to the time taken by the algorithm to convert a given plain text to cipher text. Key plays a prominent role in encryption and decryption algorithm and its size determines the strength of encryption algorithms. The increase in key size reduces the speed of the algorithm but in turn increases the security.

In this work, the main security issues and existing solutions in MANET is reviewed, particularly in which has not been widely addressed. An enhanced RSA Cryptographic Algorithm for mobile ad hoc networks is proposed. The purpose of this architecture is to keep the data confidential due to malicious activity in the route.

## II. PROBLEM STATEMENT

The main objective of this work is to achieve data confidentiality and authentication by enhanced RSA cryptographic algorithm.

An Implementation of the proposed enhanced RSA cryptographic algorithm is provided, which gives high data security with authentication.

## III. LITERATURE REVIEW

[ HYPERLINK \l "Raj14" 3 ] In his paper developed a strategy to choose one of the authenticated routing protocols according to its security-effectiveness, study it and analyze its functionality and performance. The authenticated routing for ad hoc networks secure routing protocol was chosen for analysis. He provides specific proposed solution against the different attacks in mobile Ad-hoc network.

4]**}** In their paper presents the detailed study of the popular Encryption Algorithms such as DES, AES, RSA, Diffie-Hellman, DSA and hashing algorithms. The also added that to provide more security to the network and data, different encryption methods have to be used.

[ HYPERLINK \l "YuP11" 5 ]in their work they introduced a novel, efficient and lightweight encryption protocol that fulfills the need for security protection in wireless ad-hoc networks. This protocol ensures the privacy of communication from node to node and prohibits the modification of sensitive data by dynamically changing the secret key for data encryption during packet transmission. Under the protection of this protocol, only the original sender and authorized recipient are able to decode the cipher text using the secret key that is in their possession only. Therefore, the weakness of pre-shared key encryption is overcome and other wireless attacks are prevented. Experiment results with different network configurations and key sizes have been simulated. They indicate that this i-key protocol design is efficient, with low commutation overhead, while providing better and stronger data protection compared with other common security protocols in IEEE 802.11 wireless network. Furthermore, the dynamic encryption and decryption architecture in i-key protocol is flexible; other secure systems can also adopt it as a secondary security enhancement without compromising system performance.

## IV. BASIS OF CRYPTOGRAPHY

Cryptography has a long history, but in general it is still very strange, because it is only in a small area, such as the Military, Intelligence, Diplomatic and other sensitive sectors. Computer cryptography is the study of computer information encryption, decryption and transformation of scientific, inter-disciplinary mathematics and computer, it is an emerging discipline.6]**}**

## V. RSA ALGORITHM

This algorithm was Introduced at the time when the era of electronic email was expected to soon arise, RSA implemented two important ideas:

1. Public-key encryption:This idea omits the need for a “courier" to deliver keys to recipients over another secure channel before transmitting the originally-intended message. In RSA, encryption keys are public, while the decryption keys are not, so only the person with the correct decryption key can decipher an encrypted message. Everyone has their own encryption and decryption keys. The keys must be made in such a way that the decryption key may not be easily deduced from the public encryption key.
2. Digital signatures: The receiver may need to verify that a transmitted message actually originated from the sender (signature), and didn't just come from there (authentication). This is done using the sender's decryption key, and the signature can later be verified by anyone, using the corresponding public encryption key. Signatures therefore cannot be forged. Also, no signer can later deny having signed the message.

This is not only useful for electronic mail, but for other electronic transactions and transmissions, such as fund transfers. The security of the RSA algorithm has so far been validated, since no known attempts to break it have yet been successful, mostly due to the difficulty of factoring large numbers $n = pq$, where $p$ and $q$ are large prime numbers[ HYPERLINK \l "Mil09" 7 ].

## VI. RSA ALGORITHM OPERATION

The RSA algorithm involves three steps: ***key generation***, ***encryption*** and ***decryption***.

**Key generation:** RSA involves a public key and a private key. The public key can be known by everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted in a reasonable amount of time using the private key. The keys for the RSA algorithm are generated the following way:

1. Choose two distinct prime numbers $p$ and $q$.
   - For security purposes, the integers$p$ and $q$ should be chosen at random, and should be of similar bit-length. Prime integers can be efficiently found using a primarily test.
2. Compute $n = pq$.
   - $n$ is used as the modulus for both the public and private keys. Its length, usually expressed in bits, is the key length.
3. Compute $\varphi(n) = \varphi(p)\varphi(q) = (p - 1)(q - 1) = n - (p + q - 1)$, where $\varphi$ is Euler's totient function. This value is kept private.
4. Choose an integer $e$ such that $1 < e < \varphi(n)$ *and* $\gcd(e, \varphi(n)) = 1$; i.e., $e$ and $\varphi(n)$ are co-prime.
   - $e$ is released as the public key exponent.
   - $e$ having a short bit-length and small Hamming weight results in more efficient encryption – most commonly 216 + 1 = 65,537. However, much smaller values of e (such as 3) have been shown to be less secure in some settings.
5. Determine $d$ as $d \equiv e-1 \pmod{\varphi(n)}$; i.e., $d$ is the modular multiplicative inverse of $e$ (*modulo* $\varphi(n)$).
   - This is more clearly stated as: solve for $d$ given $d \cdot e \equiv 1 \pmod{\varphi(n)}$
   - This is often computed using the extended Euclidean algorithm. Using the pseudo code in the Modular integers section, inputs a and n correspond to $e$ and $\varphi(n)$, respectively.
   - $d$ is kept as the private key exponent.

The public key consists of the modulus $n$ and the public (or encryption) exponent $e$. The private key consists of the modulus $n$ and the private (or decryption) exponent $d$, which must be kept secret. $p$, $q$, and $\varphi(n)$ must also be kept secret because they can be used to calculate $d$.

- An alternative, used by PKCS#1, is to choose $d$ matching $de \equiv 1\ (mod\ \lambda) with\ \lambda = lcm(p-1, q-1)$, where lcm is the least common multiple. Using $\lambda$ instead of $\varphi(n)$ allows more choices for $d$. $\lambda$ can also be defined using the Carmichael function, $\lambda(n)$.

Since any common factors of $(p\text{-}1)$ and $(q\text{-}1)$ are present in the factorization of $p*q\text{-}1$, it is recommended that $(p\text{-}1)$ and $(q\text{-}1)$ have only very small common factors, if any besides the necessary 28]**}**.

**Encryption:** Alice transmits her public key $(n, e)$ to Bob and keeps the private key $d$ secret. Bob then wishes to send message $M$ to Alice.

He first turns $M$ into an integer $m$, such that $0 \le m < n$ and $gcd(m, n) = 1$ by using an agreed-upon reversible protocol known as a padding scheme. He then computes the cipher text $C$ corresponding to

$$C \equiv m^e\ (mod\ n)$$

This can be done efficiently, even for 500-bit numbers, using Modular exponentiation. Bob then transmits $C$ to Alice.

Note that at least nine values of $m$ will yield a cipher text $C$ equal to $m$[ HYPERLINK \l "Wik15" 8 ].

**Decryption:** Alice can recover $m$ from $C$ by using her private key exponent $d$ via computing

$$m \equiv C^d\ (mod\ n)\}$$

Given $m$, she can recover the original message $M$by reversing the padding scheme [8].

## VII. PROPOSED WORK

### Data Encryption and Authentication

In order to have data security, all the data packets are encrypted and decrypted using a private key and authentication can be obtained by asymmetric cryptography [9]. For this, an enhanced RSA cryptographic algorithm is proposed.

**Enhanced RSA cryptographic algorithm (ERSA) –** RSA algorithm is the system used to encrypt with one key and decrypt with another, in other words we say we can use one public key to encrypt and a private key to decrypt, and the inverse is also true, that is to say; a private key to encrypt and a public key to decrypt. Therefore the inverse function is valid here. E.g.:

$$1/_2 \times 2/_1 = 1$$

Assuming we have a secrete number 7 and we want to multiply it with another number say 328 i.e. $7 \times 328 = 2296$, the result is a large number and it is not clear that 7 is part of this number it is said to be hidden or encrypted. But if we multiply the $2296 \times 1/_{328} = 7$, we will still have our secrete number back.

Therefore, the relation used for encryption is given as: $M^e\,mod\ n = C$. Similarly, the relation used for decryption is given as: $C^d mod\ n = M$, where $C$ is the cypher text, $M$ is the original message, $e$, $n$ are the public key that can be shared with any one, and $d$ is the private key that must be kept secret.

In other words, RSA algorithm can be called ***a one-way trapdoor function,*** meaning that you cannot undo the encryption function without knowing the ***trapdoor***, and this is nothing rather than the number ***n.*** By increasing the strength and complexity of ***n,*** we will tend to arrive at a very strong cypher text that will be difficult and consume a lot of time to decrypt.

Initially, ***n*** is given as the product of two large prime numbers *p and q*. It is so difficult for a third party to find the value of ***n,*** because of prime factorization. Actually it is easy to find the product of two or three prime numbers e.g. $3 \times 5 = 15$. Similarly, it is easy to find the product of two or more large prime numbers e.g. $1889 \times 3547 = 6700283$. But it is obviously very had to take a very large product of two or more prime numbers and factor it to find out the constituting prime numbers e.g.: $Prime1 \times Prime2 = 6700283$.

Hence, this is the main problem that supposed to be impossible for the world computing resources to solve for many decades (50 years is probably a law estimate). Therefore, knowing the factors of ***n*** is the ***trapdoor*** in this trapdoor function, that is to say, if you know the factors of ***n,*** then you can decrypt a message that has been encrypted with its public key.

Another important aspect is generator function $\varphi(n)$, which was initially given as:

$$\varphi(n) = (p-1)(q-1)$$

This is very important to us to drive the inverse of ***e.*** $\varphi(n)$ has a limitation that: it must not share a factor with ***e.*** This must be checked before using the algorithm.

To ensure the proper complexity of RSA algorithm, a new coefficients ***r*** and ***s*** have been added to the generator function $\varphi(n)$ as follow:

$$\varphi(n) = (p-1)(q-1)(r-1)(s-1)$$

And then complete all the steps, ***r*** and ***s*** are taken in to consideration. This will increase the complexity of the trapdoor ***n*** as new additional large prime numbers area added ***r*** and ***s***, which in turn become more difficult and time consuming to decrypt as the value of ***n*** has been increased.

The flowchart below obviously explains the enhanced algorithm of RSA which is adopted here.

**Example:**

We do an RSA encryption with artificially small parameters.

Key generation, Entity A chooses the primes p = 7, q =11, r=13 and s=17, And computes $n = p.q.r.s = 17017\ and \varphi(n) = (p-1)(q-1)(r-1)(s-1)$ =11520.

A chooses e =79, find d =319 such that e*d mod $\varphi$(n)=1. A's public key pair is (n = 17017, e =79), while A's private key is d =319.

**Encryption:** To encrypt a message m = 52, B uses an algorithm for modular exponentiation 2 to compute

$$C = M^e \ mod\ n = 52^{79}\ mod\ 17017 = 4693$$

And sends this to A.

**Decryption:** To decrypt C, A computes

$$M = C^d\ mod\ n = \ 4693^{319}\ mod\ 17017 = 52$$

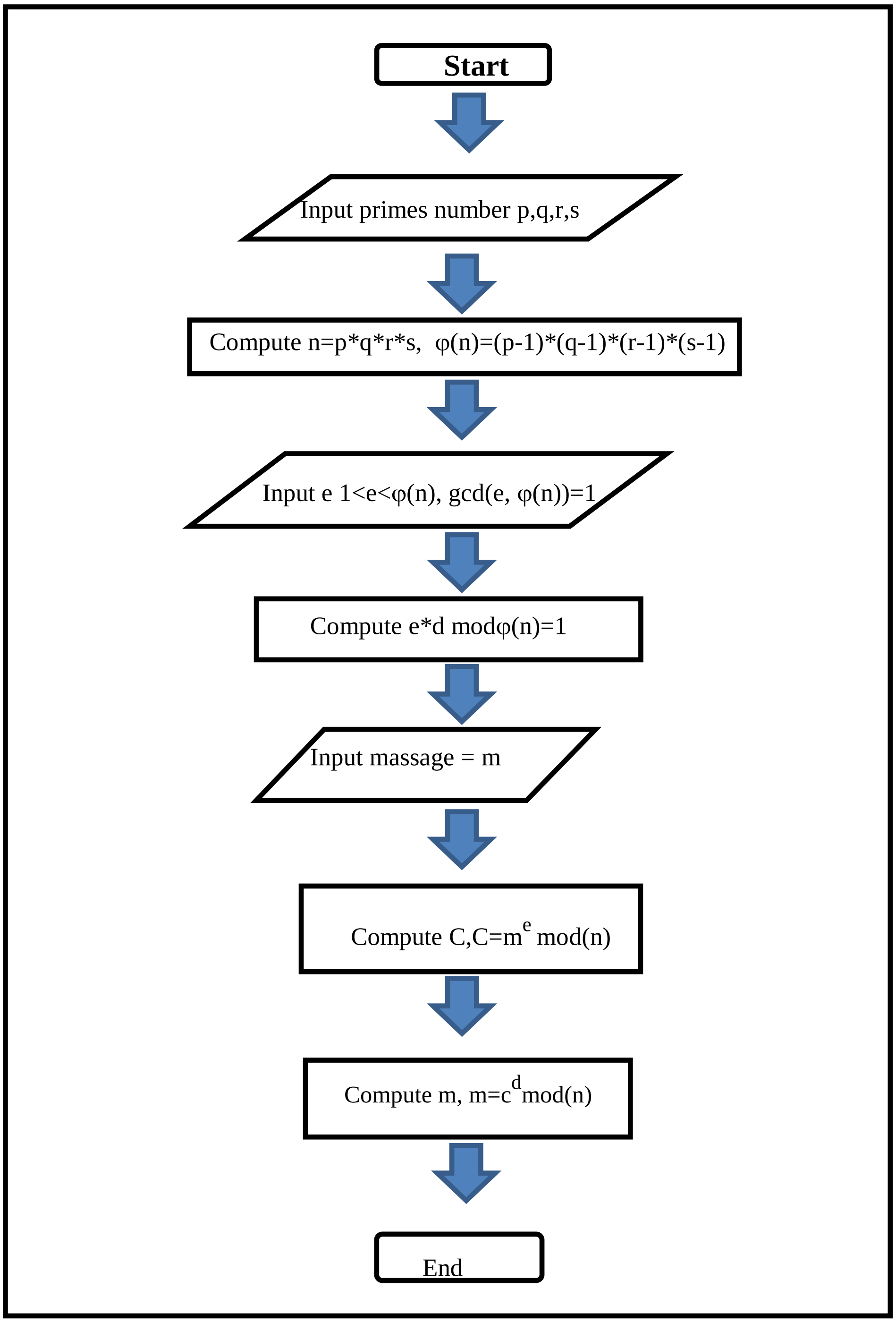

Start
Input primes number p,q,r,s
Compute n=p*q*r*s, φ(n)=(p-1)*(q-1)*(r-1)*(s-1)
Input e 1<e<φ(n), gcd(e, φ(n))=1
Compute e*d modφ(n)=1
Input massage = m
Compute C,C=m^e mod(n)
Compute m, m=c^d mod(n)
End

**Figure 1: Enhanced RSA Algorithm execution flow chart**

## VIII. IMPLEMENTATION AND ANALYSIS OF THE ENHANCED RSA ALGORITHM (ERSA)

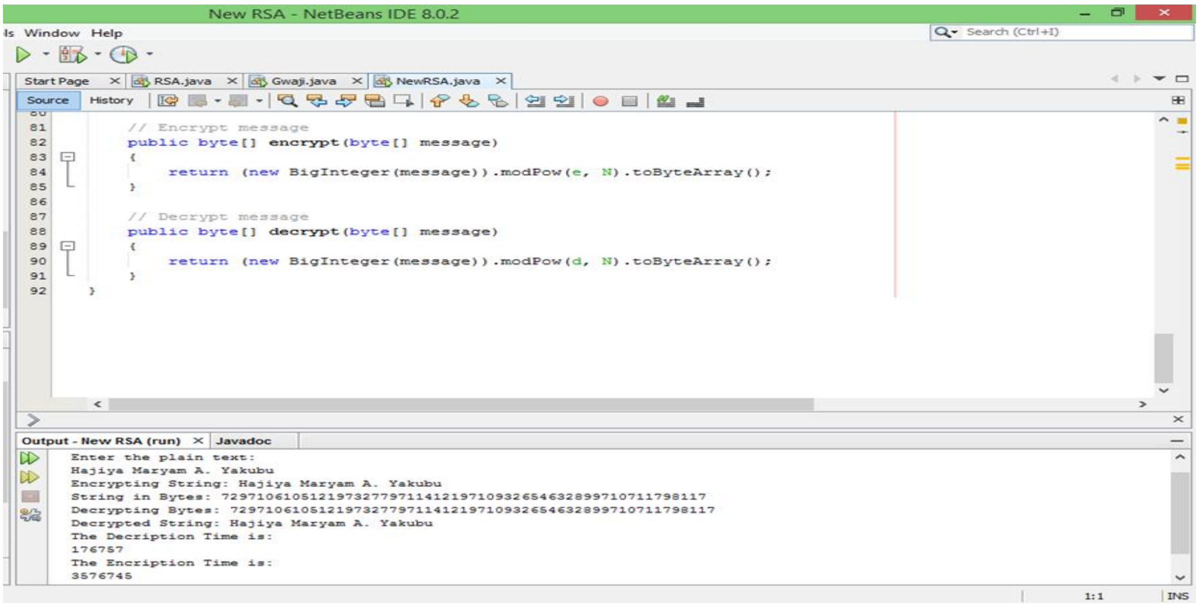
The proposed algorithm is implemented using Java, where ten (10) different scenarios where created using ten (10) different Message texts but using same code. Each message is encrypted and decrypted, and the time taking to do so is also measured in Nano seconds.

To take this time reading in JAVA, a command known as System.nanoTime() is used, which return the current value of the most precise available system timer in Nano second.

Nano second is a thousand million for the second. Therefore, it's a very small unit of time.

By running the program in different time, the value of the timer may change depending on what the rest of the computer is doing at that instance of time, since the operating system does many things beside this program work at that instance.

In order to compare the two algorithms, the above method is repeated using the old algorithm, and analysis is provided using bar chart created from the result generated in each case using ten (10) different message texts. The result is tabulated in the table below:



```
        // Encrypt message
        public byte[] encrypt(byte[] message)
        {
            return (new BigInteger(message)).modPow(e, N).toByteArray();
        }

        // Decrypt message
        public byte[] decrypt(byte[] message)
        {
            return (new BigInteger(message)).modPow(d, N).toByteArray();
        }
    }
```

```
Enter the plain text:
Hajiya Maryam A. Yakubu
Encrypting String: Hajiya Maryam A. Yakubu
String in Bytes: 7297106105121973277971141219710932654632899710711798117
Decrypting Bytes: 7297106105121973277971141219710932654632899710711798117
Decrypted String: Hajiya Maryam A. Yakubu
The Decription Time is:
176757
The Encription Time is:
3576745
```

**Figure 2: Program feature**

**Table 1: Encryption and Decryption time comparison in Nano Seconds**

| Scenarios | Messages | Old RSA encryption time (ns) | Old RSA decryption time (ns) | New RSA encryption time (ns) | New RSA decryption time (ns) |
|---|---|---|---|---|---|
| 1 | Sharda University, India | 1506988 | 130001 | 4467938 | 188730 |
| 2 | Kano University of science and Tech. | 1607910 | 134563 | 4048856 | 261143 |
| 3 | Senator Rabiu Musa Kwankwaso | 1371855 | 62150 | 4403508 | 217239 |
| 4 | Bello Musa | 1445979 | 84387 | 4236445 | 91229 |

| | | | | | |
|---|---|---|---|---|---|
| | Yakubu | | | | |
| 5 | M. Tech. Computer Sci | 3210690 | 80965 | 3225514 | 146536 |
| 6 | RSA algorithm work | 1951160 | 84957 | 3310471 | 114036 |
| 7 | Computer Network security | 1961993 | 108334 | 4692589 | 176186 |
| 8 | Dr. Abdullahi Umar Ganduje | 1598787 | 94080 | 3607535 | 198423 |
| 9 | Hajiya Maryam A. Yakubu | 1407776 | 123159 | 3576745 | 176757 |
| 10 | Masters final Work Completed | 1156327 | 105483 | 4745046 | 206406 |

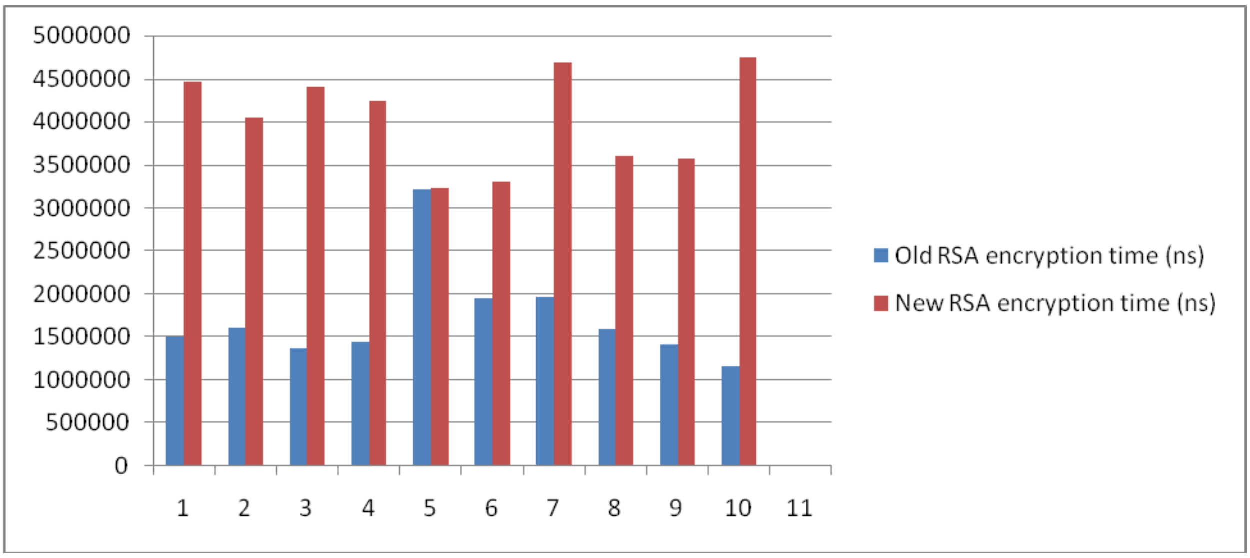


**Figure 3: Old RSA vs. Enhanced RSA Encryption Time (ns)**

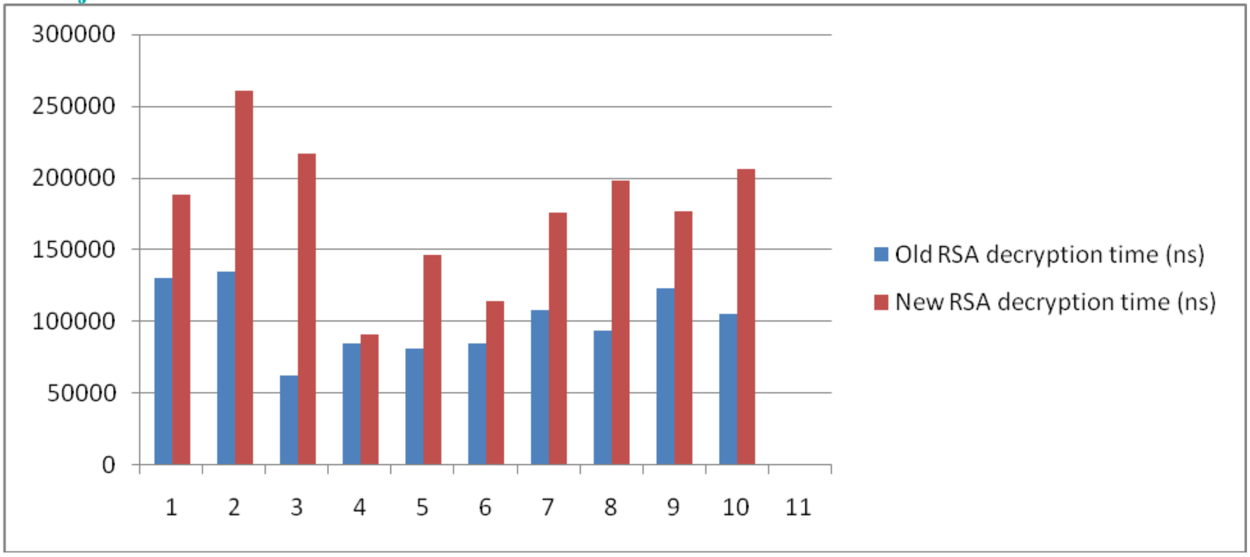


Figure 4: Old RSA vs. Enhanced RSA Decryption Time (ns)

## IX. DISCUSSION AND EVALUATION OF THE ENHANCED RSA ALGORITHM

As we can see, by increasing the parameters constituting the generator function $\varphi(n)$ will make it to be more difficult instead of two parameters as in old RSA algorithm. The increase in the degree of that function by two, leads to change in all of the concerning equations of cryptography and their attacks. It is clear that, the modification of algorithm presents a new strategy which is different from that of classical.

When the enhanced algorithm is threatened during attacks process which may related directly by factorial multiplication properties, there is need to increase it by two the number of probabilities than that of old RSA algorithm such that in old RSA system the attack takes a factorial of two variables which are p and q only, however at the enhanced algorithm, it takes factorial of four variables which are p, q, r and s into consideration. Therefore, more time will be needed to break the enhanced algorithm.

## X. CONCLUSION

From the result of the implementation of the enhanced RSA algorithm, we can quickly drive that, time taking to encrypt or decrypt a message is far more than time taking to do so in the old RSA, hence, this bring us to a very important conclusion that the aim of the enhanced RSA is achieved in respect to encryption and decryption time complexity using the same method in the comparative analysis of the two algorithms.

This new approach can enhance the productivity and security of the information and communication technology especially in commercial sector.

Lastly, the modification can be further enhanced in other to be used in more field of life such as military.

## REFERENCES

[1] Diaa Salama, Hatem Hatem Abdual Kader, and Mohiy Hadhoud, "Studying the effects of Most Common Encryption Algorithms," *International Arab Journal of e-Technology*, vol. Vol.2, No.1, January 2011.

[2] A. L. Jeeva, Dr. V. Palanisamy, and K. Kanagaram, "Comparative analysis of performance efficiency and security measures of some encryption algorithms," *International Journal of Engineering Research and Applications(IJERA)* , vol. Vol.2, no. Issue 3, pp. 3033-3037, May-June 2012.

[3] Vishal Rajput, "Authenticated Data Transmission in Decentralized Wireless Mobile Ad-Hoc Network (MANET)," *International Journal of Applied Engineering Research*, vol. Volume 9, no. Number 20, pp. 6707-6714, 2014.

[4] Deepti Ranaut and Madal Lal, "A Review on Security Issues and Encryption Algorithms in Mobile Ad-hoc Network," *International Journal of Science and Research (IJSR)*, vol. 3 , no. 6, pp. 146-148, June 2014.

[5] Peter H. Yu and Udo W. Pooch, "Security and Dynamic Encryption System in Mobile Ad-Hoc Network Mobile Ad-Hoc Networks: Protocol Design, Prof. Xin Wang (Ed.)," 2011.

[6] Na Qi et al., "Analysis and Research of RSA Algorithm," *Information Technology Journal*, pp. 1818-1824, 2013.

[7] Evgeny Milanov, "The RSA Algorithm," June 2009.

[8] Wikipedia. (2015, August) Wikipedia, the free encyclopedia. [Online]. https://en.wikipedia.org/wiki/RSA_(cryptosystem)

[9] W. Stallings, *Cryptography and network security 4th Edition*. prentice hall, 2005.