International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8, Issue-1, May 2019

# Enhancing the Data Security by using RSA Algorithm with Application of Laplace Transform Cryptosystem

**G. Nagalakshmi, A. Chandra Sekhar, N. Ravi Sankar, K. Venkateswarlu**

***Abstract:*** ***Encryption and Decryption schemes based on applications of Laplace Transforms are unable to provide more security to communicate the information. Rivest, Shemir, Adleman (RSA) Cryptosystem is popular public-key algorithm. This paper provides the conditions that give rise to the RSA Cryptosystem based on the Laplace Transform techniques. The modified RSA cryptosystem is explained with an algorithm. The proposed algorithm is implemented using a high level program and time complexity of the proposed algorithm is tested with RSA cryptosystem algorithms. The comparison reveals that the proposed algorithm enhances the data security as compare with RSA cryptosystem algorithms and application of Laplace Transform for cryptosystem scheme. The statistical analysis for the proposed and existing algorithms is provided***

***Index Terms— Laplace Transform Cryptography; Secret key; Public key; RSA cryptosystem.***

## I. INTRODUCTION

Several cipher algorithms have been used for Cryptographic with the Application of Laplace Transform [4,6,8]. However, the security of the proposed algorithms for several designs has been demonstrated using statistical test results [7]. Nagalakshmi et al., [3] introduced a new scheme in Cryptography whose construction is based on the application of Laplace Transform encrypt a sting by using series expansion of f(t). Hiwarekar [4] developed an application of Laplace Transform for Cryptography scheme. Various Laplace Transform papers for Cryptography are found in literature [9,10]. Gupta and Mishra [5] posit that the single-iteration procedure offers a weak encryption scheme by showing that the cypertext messages can be decrypted by elementary modular arithmetic. In the end of the paper concludes that the single-iteration Hiwarekar application of Laplace transform for Cryptographic process is a weak scheme.

**Revised Manuscript Received on 30 May 2019.**
***** Correspondence Author
**G. Nagalakshmi,** Assistant Professor**,** Dept of Applied Mathematics, GIS, GITAM (Deemed to be University), Andhra Pradesh, India.
**Prof. A. Chendra Sekhar,** Dept of Applied Mathematics, GIS, GITAM (Deemed to be University), Andhra Pradesh, India.
**Prof. N. Ravi Sankar,** Dept of Applied Mathematics, GIS, GITAM (Deemed to be University), Andhra Pradesh, India.
**K. Venkateswarlu,** Assistant Professor, Dept of Computer Science, GIT, GITAM (Deemed to be University), Andhra Pradesh, India.



The RSA Cryptosystem is the most widely – used public key cryptography algorithm in the world. It can be used to encrypt a message without the need to exchange a secret key separately. RSA is an algorithm used by modern computers to encrypt and decrypt messages. Various RSA papers for cryptography are found in literature [2]. Here in this work we use RSA Algorithm with application of Laplace Transform (RSALT) for cryptosystem.

### A. RSA Public key Cryptography (RSA):

RSA algorithm involves two keys (1) **public key** (2) **private key.** The public key used for encrypting messages and messages can be decrypted using the private key. The keys are generated as following [2]

1. Randomly select two different prime numbers **p** and **q**.
2. Find **n = p*q.**
3. Find **φ(n) = (p – 1)(q – 1)**
4. Select an integer *e* such that **1 < e < φ(n)** and Greatest Common Divisor of (***e***, **φ(*n*)) = 1.**
5. Evaluate $d = e^{-1} \bmod \varphi(n)$
6. {e, n} is Public Key
7. {d, n} is Private Key

### B. Laplace Transform Cryptography (LT):

Laplace transform has many applications in various fields such as a simple Laplace Transform is conducted while sending signals over any two-way communication medium (FM/AM stereo, 2-way radio sets, cellular phones).When information is sent over medium such as cellular phones, they are first converted into time-varying wave, and then it is super-imposed on the medium. In this way, the information propagates.

**C. Definition:** If f (t) is a function defined for all positive values of t, then the Laplace transform of f (t) is defined as

$$L\{f(t)\} = \int_0^\infty e^{-st} f(t)dt = F(s) ----(1)$$

provided that the integral exists. Here the parameter‘s‘is a real or complex number [1].

The corresponding inverse Laplace transform is

$$L^{-1}\{F(s)\} = f(t) ----(2)$$

**D. Linear properties:** Laplace transform is a linear transform.

If $L\{f_1(t)\} = F_1(s)$,

$L\{f_2(t)\} = F_2(s)$, …, $L\{f_n(t)\} = F_n(s)$, then

$$L\{c_1 f_1(t) + c_2 f_2(t) + \ldots + c_n f_n(t)\}$$
$$= c_1 F_1(s) + c_2 F_2(s) + \ldots + c_n F_n(s) ----(3)$$

Where $c_1, c_2 \ldots, c_n$ are constants

**E. Some standard formulas:**

* If $L(t^n) = n!/s^{n+1}$ then $L^{-1}(n!/s^{n+1}) = t^n$ ---(4)
* If $L(e^{at}) = 1/s\text{-}a$ then $L^{-1}(1/s\text{-}a) = e^{at}$ ---(5)
*If $L(\sinh at)^2$ then $L^{-1}(a/s^2 - a^2) = \sinh at$ ---(6)
* $L(t^n e^{at}) = n!/(s\text{-}a)^{n+1}$ then $L^{-1}(n!/(s\text{-}a)^{n+1}) = t^n e^{at}$---(7)

Where a = 1, 2, 3, · · · , the positive integers [1].

## II. PROPOSED ALGORITHM

The proposed RSA algorithm with application of Laplace Transform (RSALT) the proposed work main intention work is to improve the date security and to eliminate the redundant calculation in encryption. In this work we use to a method for the cryptosystem.

**A. Encipher algorithm:**

**Step 1**: First level of encryption select the plain text $P_0, P_1, \ldots, P_m$, in ASCII code integer $M_0, M_1, \ldots, M_m$,

where $M_i \in N$

**Step2:** Generate two large random primes' p and q of approximately equal size.

**Step 3**: Generate the function f (t) using combination of two primes p and q.

i.e., $$f(t) = \sum_{i=0}^{m} \frac{q^i t^{i+1}}{((p+i)-1)!}$$ or

$\sum_{i=0}^{m} q^i e^{t(p+i)}$ or $\sum_{i=0}^{m} q^i \sinh(p+i)t$ and so many.

**Step 4**: Let consider the function

$$f(t) = \sum_{i=0}^{m} \frac{q^i t^{p+i}}{((p+i)-1)!} \quad \text{-------- (8)}$$

and Calculate

$$\sum_{i=0}^{m} \frac{M_i q^i t^{p+1}}{((p+i)-1)!} = \sum_{i=0}^{m} \frac{N_i t^{p+1}}{((p+i)-1)!} \quad \text{-------(9)}$$

where $N_i = M_i * q^i$ and $0 \le i \le m$

**Step 5**: Calculate the Laplace Transform of

$$L\left\{\sum_{i=0}^{m} \frac{N_i t^{p+1}}{[(p+i)-1]!}\right\} = \sum_{i=0}^{m} L\left\{\frac{N_i t^{p+1}}{[(p+i)-1]!}\right\}$$

$$= \sum_{i=0}^{m} \frac{R_i}{s^{(p+i)+1}} where R_i = L\{N_i t^{p+1}\} - - - 10$$

-----(10)

Where **S** being a non zero real or complex number.

**Step 6**: At second level of encryption each integer of $R_0, R_1, \ldots, R_i$ is encrypted by the RSA Algorithm.

**Step 7**: Now calculate n = p*q and φ(n) =(p-1)*(q-1).

**Step 8:** Choose an integer *e*, 1 < e < φ (n), such that gcd (e, φ (n)) = 1

**Step 9**: Find the secret exponent *d*, 1 < d < φ (n), such that e*d ≡ 1 (mod φ (n)).

**Step10**: The {n, e} is public key
and {f(t), d, n} is private key
Keep all the secret values of d and φ(n).

**Step 11**: Calculate ciphertext $C = R^e$ mod n then get integer of ciphertext $C_0, C_1, \ldots, C_m$.

**Step 12**: Each integer of ciphertext $C_0, C_1, \ldots, C_m$ is converted to its construct by ASCII character are stored as the ciphertext C.

**B. Decipher algorithm:**

**Step 1**: First level of decryption the cipher text C converted to ASCII values of $C_0, C_1, \ldots, C_m$.

**Step 2**: Each integer of $C_0, C_1, \ldots, C_m$ is converted into RSA algorithm $m = C^d$ mod n.

**Step 3**: Get "$m_0, m_1, \ldots, m_m$" in the above method. Now calculate $R_i = m_i + nK_i$ with number of multipliers of mod n is $K_i$.

**Step 4:** Now get $R_0, R_1, \ldots, R_m$ and construct $\sum_{i=1}^{m} \frac{R_i}{s^{(p+i)+1}}$ --(11)

**Step 5:** Apply inverse Laplace transform

$$L^{-1}\{\sum_{i=0}^{m} \frac{R_i}{s^{(p+i)+1}}\} = \sum_{i=0}^{m} \frac{N_i t^{p+1}}{[(p+i)-1]!}\} \quad \text{---- (12)}$$

$$\Rightarrow \sum_{i=0}^{m} \frac{N_i t^{p+1}}{[(p+i)-1]} = \sum_{i=0}^{m} \frac{M_i q^i t^{p+1}}{[(p+i)-1]!}$$

where $N_i = M_i * q^i$ and $0 \le i \le m$.

**Step 6**: Each integer $M_i$ are converted to their corresponding ASCII code values are $M_0, M_1, \ldots, M_m$ and hence get the original plain text is $P_0, P_1, \ldots, P_m$.

Public key = {n, e}
Private key = {f(t), d, n}

## III. EXAMPLE

In this section we are presented the example for method of Encryption and method of decryption.

**A. Method of Encryption:**

**Step 1:** First level of encryption select the plain text $P_0, P_1, \ldots, P_m$ and convert into ASCII code values $M_0, M_1, \ldots, M_m$. For example the given plain text is "AP 31". Here i = 5.

Based on the ASCII code table the plain text allocated A = 65, P = 80, space = 32, 3 = 51, 1 = 49.

The plain text finite sequence is $M_0 = 65$, $M_1 = 80$, $M_2 = 32$, $M_3 = 51$, $M_4 = 49 \ldots M_i = 0$ for $i \ge 6$.

**Step 2:** Generate two large primes numbers p = 11, q = 13.

**Step 3:** Generate the function f (t) using combination of two random primes p and q (8).

i.e., $$f(t) = \frac{q^0 t^{p+0}}{((p+0)-1)!} + \frac{q^1 t^{p+1}}{((p+1)-1)!} + \frac{q^2 t^{p+2}}{((p+2)-1)!}$$

$$+ \ldots\ldots\ldots + \frac{q^i t^{p+i}}{((p+i)-1)!} = \sum_{i=0}^{m} \frac{q^i t^{p+i}}{((p+i)-1)!} by(8)$$

**Step 4:** Let us consider f (t) = $\sum_{i=0}^{m} \frac{M_i q^i t^{p+i}}{((p+i)-1)!}$.

$$\Rightarrow f(t) = \frac{M_0 q^0 t^{p+0}}{((p+0)-1)!} + \frac{M_1 q^1 t^{p+1}}{((p+1)-1)!} + \frac{M_2 q^2 t^{p+2}}{((p+2)-1)!}$$

$$+ \frac{M_3 q^3 t^{p+3}}{((p+3)-1)!} + \frac{M_4 q^4 t^{p+4}}{((p+4)-1)!} by(9)$$

$$\Rightarrow f(t)=\frac{65(13)^0 t^{11+0}}{((11+0)-1)!}+\frac{80(13)^1 t^{11+1}}{((11+1)-1)!}+\frac{32(13)^2 t^{11+2}}{((11+2)-1)!}$$

$$+\frac{51(13)^3 t^{11+3}}{((11+3)-1)!}+\frac{49(13)^4 t^{11+4}}{((11+4)-1)!}----(13)$$

$$f(t)=65\frac{t^{11}}{10!}+1040\frac{t^{12}}{11!}+5408\frac{t^{13}}{12!}.$$

$$+112047\frac{t^{14}}{13!}+1399489\frac{t^{15}}{14!}$$

**Step 5:** Take Laplace transform of a polynomial on both sides

$$L\{f(t)\}=L\{65\frac{t^{11}}{10!}+1040\frac{t^{12}}{11!}+5408\frac{t^{13}}{12!}+112047\frac{t^{14}}{13!}+1399489\frac{t^{15}}{14!}\}$$

$$=65L\{\frac{t^{11}}{10!}\}+1040L\{\frac{t^{12}}{11!}\}+5408L\{\frac{t^{13}}{12!}\}+112047L\{\frac{t^{14}}{13!}\}+1399489L\{\frac{t^{15}}{14!}\}.$$

$$=\frac{65}{10!}.\frac{11!}{s^{12}}+\frac{1040}{11!}.\frac{12!}{s^{13}}+\frac{5408}{12!}\frac{13!}{s^{14}}+\frac{112047}{13!}\frac{14!}{s^{15}}+\frac{1399489}{14!}\frac{15!}{s^{16}}$$

$$=\frac{715}{s^{12}}+\frac{12480}{s^{13}}+\frac{70304}{s^{14}}+\frac{1568658}{s^{15}}+\frac{20992335}{s^{16}}.$$

$$L\{f(t)\}=\frac{R_0}{s^{12}}+\frac{R_1}{s^{13}}+\frac{R_2}{s^{14}}+\frac{R_3}{s^{15}}+\frac{R_4}{s^{16}}\ by(10).$$

Resulting values are $R_0$ = 715, $R_1$ = 12480, $R_2$ = 70304, $R_3$ = 1568658, $R_4$ = 20992335. $R_i$ = 0 for $i \geq 6$

**Step 6:** At second level of encryption each integer of $R_1, R_2, \ldots, R_i$ is encrypted by the RSA algorithm.

**Step 7:** compute n = p * q = 11 * 13 = 143 and $\phi(n)=(p-1)*(q-1)$ = (11-1) * (13-1) =10 * 12 = 120.

**Step 8:** Choose an integer e = 7, 1 < e < 120, such that greatest common division (7, 120) = 1

**Step 9**: Find the secret exponent d = 103, 1 < d < 120, such that e*d ≡ 1 (mod 120).

**Step10**: The public key is {n, e}
and the secret key is { f (t), d, n}.
Keep all the values are d and φ(n).

**Step 11**: RSA Encryption form is $C = R^e$ mod n.

(i) $C_0 = (R_0)^e$ mod n.
$C_0 = (715)^7$ mod 143 = 143.

(ii) $C_1 = (R_1)^e$ mod n.
$C_1 = (12480)^7$ mod 143 = 52.

(iii) $C_2 = (R_2)^e$ mod n.
$C_2 = (70304)^7$ mod 143 = 130.

(iv) $C_3 = (R_3)^e$ mod n
$C_3 = (1568658)^7$ mod 143 = 130.

(v) $C_4 = (R_4)^e$ mod n
$C_4 = (20992335)^7$ mod 143 = 78.

The encrypted cipher text values are $C_0$ = 143, $C_1$= 52, $C_2$ = 130, $C_3$ = 130, $C_4$ =78.

**Step 12**: Each integer of cipher text $C_1, C_2, \ldots, C_5$ is converted to its constructed by ASCII character are stored as the cipher text C is "**Å4ëëN** ".

**B. Method of Decryption**:

**Step 1**: First level of decryption the cipher text "**Å4ëëN**" converted to ASCII values of $C_0, C_2, \ldots, C_m$,
i.e, $C_0$ = 143, $C_1$= 52, $C_2$ = 130, $C_3$ = 130, $C_4$ = 78.

**Step 2**: Convert the given cipher text by RSA algorithm.
Here $C_0$ = 143, $C_1$= 52, $C_2$ = 130, $C_3$ = 130, $C_4$ = 78.

(i) $m_0 = (C_0)^d$ mod n
= $(143)^{103}$ mod 143 = 143

(ii) $m_1 = (C_1)^d$ mod n
= $(52)^{103}$ mod 143 = 39

(iii) $m_2 = (C_2)^d$ mod n
= $(130)^{103}$ mod 143 = 91

(iv) $m_3 = (C_3)^d$ mod n
= $(130)^{103}$ mod 143 = 91

(v) $m_4 = (C_4)^d$ mod n
= $(78)^{103}$ mod 143 = 78

**Step 3**: Get "$m_0, m_1, \ldots, m_m$" in the above method. Now calculate $R_i = m_i + nK_i$ with number of multipliers of mod n is $K_i$ where i= 0, 1,…,m.

(i) $R_0 = m_0 + K_0$ n = 143 + 0 (143) = 143
(ii) $R_1 = m_1 + K_1$ n = 39 + 87 (143) = 12480
(iii) $R_2 = m_2 + K_2$ n = 91 + 491 (143) = 70304
(iv) $R_3 = m_3 + K_3$ n = 91 + 10969 (143) = 1568658
(v) $R_4 = m_4 + K_4$ n = 78 + 146799 (143) = 20992335

**Step 4:** Now we get $R_0, R_1, \ldots, R_m$ and construct by using equation(14). $\sum_{i=0}^{m}\frac{R_i}{s^{(p+i)+1}}=\frac{R_0}{s^{12}}+\frac{R_1}{s^{13}}+\frac{R_2}{s^{14}}+\frac{R_3}{s^{15}}+\frac{R_4}{s^{16}}$

**Step 5**: Now apply inverse Laplace transform of the function $f(t)=L^{-1}\{\frac{R_0}{s^{12}}+\frac{R_1}{s^{13}}+\frac{R_2}{s^{14}}+\frac{R_3}{s^{15}}+\frac{R_4}{s^{16}}\}$.

$$\Rightarrow f(t)=L^{-1}\{\frac{143}{s^{12}}+\frac{12480}{s^{13}}+\frac{70304}{s^{14}}+\frac{1568658}{s^{15}}+\frac{20992335}{s^{16}}\}$$

$$\Rightarrow f(t)=143L^{-1}\left\{\frac{1}{s^{12}}\right\}+12480L^{-1}\left\{\frac{1}{s^{13}}\right\}+70304L^{-1}\left\{\frac{1}{s^{14}}\right\}$$

$$+156858L^{-1}\left\{\frac{1}{s^{15}}\right\}+20992335L^{-1}\left\{\frac{1}{s^{16}}\right\}$$

$$\Rightarrow f(t)=143\frac{t^{11}}{11!}+12480\frac{t^{12}}{12!}+70304\frac{t^{13}}{13!}+1568658\frac{t^{14}}{14!}+20992335\frac{t^{15}}{15!}$$

$$\Rightarrow f(t)=65(13)^0\frac{t^{11}}{10!}+80(13)^1\frac{t^{12}}{11!}+32(13)^2\frac{t^{13}}{12!}+51(13)^3\frac{t^{14}}{13!}+49(13)^4\frac{t^{15}}{14!}$$

for m by using equation(12).

**Step 6**: Consider the co-efficient of a polynomial as finite sequence is $M_0$ = 65, $M_1$ = 80, $M_2$ = 32, $M_3$ = 51, $M_4$ = 49...$M_i$ = 0 for $i \geq 6$. Each integer $M_i$ are converted to their corresponding ASCII code values are producing the original plain text is $P_0, P_1, \ldots, P_m$ obtained as "**AP 31**".

## IV. IMPLEMENTATION PLATFORM

The algorithm is implemented utilizing Visual Studio 2010 of Microsoft SQL 2008 devices on windows 7 operating system. The **.net** program is used to implement the algorithm. The used system has 32-bit operating system with Pentium (R) processor having speed 2.40 GHz, RAM of 2 GB.

### A. Testing and Analysis:

In this proposed algorithm we present frequency testing, Encryption time, Decryption time and statistical analysis. We are showed that the graph of application of Laplace Transform, RSA algorithm, proposed work (RSALT) and also compared with the each other. In statistical analysis we used LT, RSA and proposed method (RSALT) of correlation coefficient.

### B. Comparison of LT, RSA and RSALT algorithms:

Figure 1, 2, 3,4 shows the comparison of characters in cipher text for LT, RSA algorithm and proposed method of RSALT. From the below figures shown that the proposed method RSALT provides well require security compare with other algorithm.

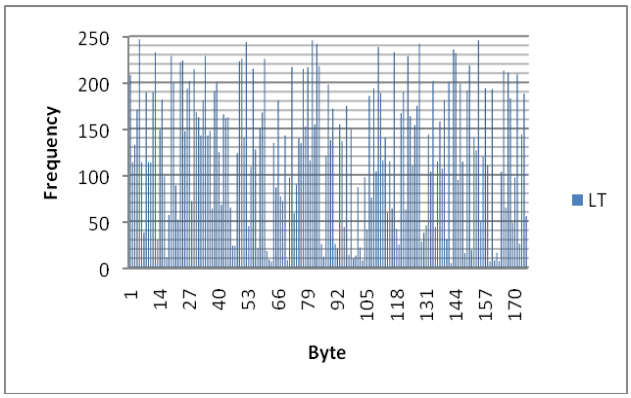


**Fig 1: Frequency of characters in ciphertext using application of Laplace transform for cryptography**

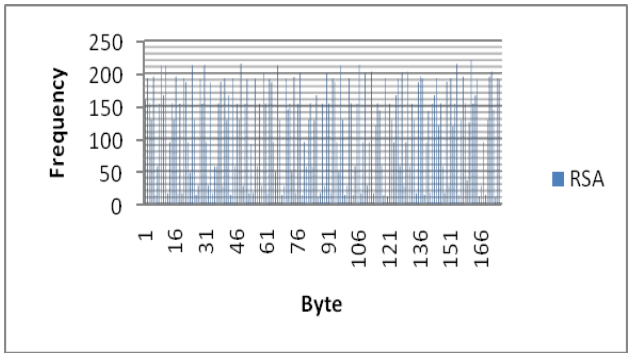


**Fig 2 : Frequency of characters in ciphertext using RSA algorithm**

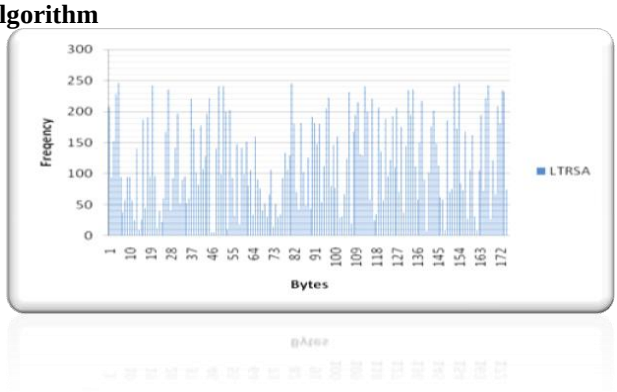


**Fig 3 : Distribution of characters in ciphertext using RSA algorithm with application of Laplace Transform (RSALT)**

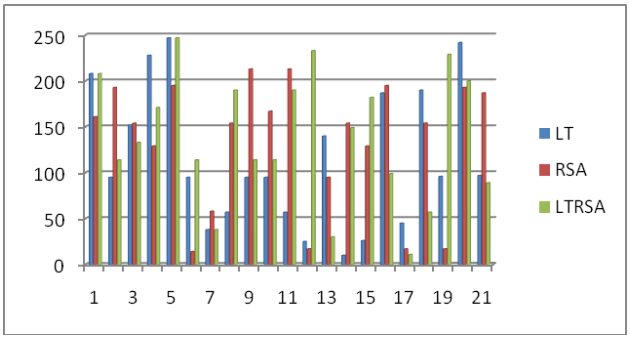


**Fig 4 : Characters distribution in cipher file with application of Laplace transform, RSA algorithm and RSALT**

### C. Encipher and Decipher Time:

Encipher and Decipher time with respect to different file sizes have been showed in table 1 and table 2. The corresponding graph as presented in Figure 5 and Figure 6. It is revealed that LT, RSA algorithm and proposed RSALT provide similar Encryption and Decryption time. Encryption and Decryption time has been calculated in milliseconds which is 1000s.

**Table 1 : File size with Time**

| File name | File Size (in Bytes) | Encryption Time (in millisec) | | |
|---|---|---|---|---|
| | | LT | RSA | RSALT |
| Test – I | 50 | 70 | 50 | 80 |
| Test – II | 100 | 140 | 110 | 140 |
| Test– III | 150 | 220 | 170 | 210 |
| Test– IV | 200 | 300 | 250 | 300 |
| Test – V | 300 | 470 | 440 | 460 |

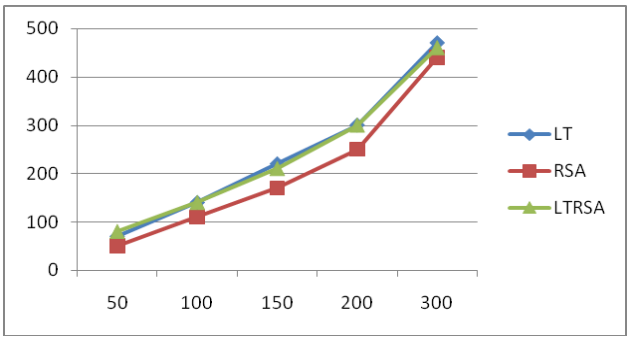


**Fig 5: Encryption Time (millisec) v/s File Size**

**Table 2 : File size with Time**

| Filename | File Size (in Bytes) | Decryption Time (in millisec) | | |
|---|---|---|---|---|
| | | LT | RSA | RSALT |
| Test – I | 50 | 123 | 161 | 140 |
| Test – II | 100 | 136 | 210 | 188 |
| Test – III | 150 | 247 | 260 | 276 |
| Test – IV | 200 | 343 | 334 | 342 |
| Test – V | 300 | 551 | 392 | 365 |

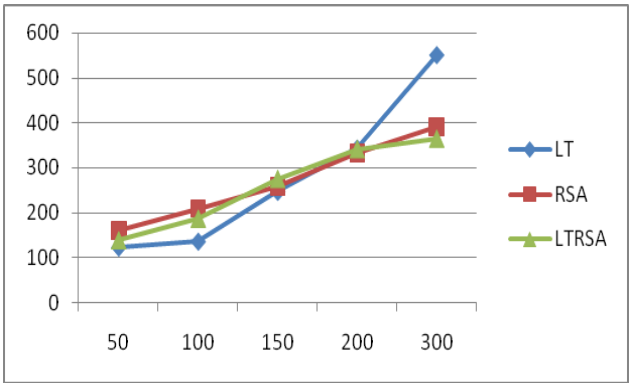


**Fig 6: Decryption Time (Millisec) v/s File size**

### C. Encrypted and Decrypted File Size Comparison:

We has been select ranging of the file sizes 50 to 450 of the plaintext then we get same file sizes ciphertext but we get to the time analysis is small difference between plaintext and ciphertext. Which has been presented in Fig 7.

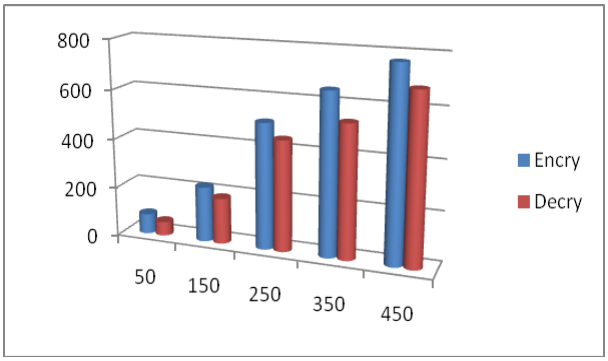


**Fig 7: Plain texts file size v/s Cipher File size**

### D. Statistical Analysis:

**Correlation coefficients** are used in statistics to measure how strong a relationship is between two pair of variables. The proposed algorithm to examine and strongly resists statistical attacks. So, cipher text and plaintext should be completely different. The correlation shows associations between the pair of values. If the correlation Coefficient equals one, that means the plaintext and its encryption is identical. If the correlation coefficient equals zero, that means the ciphertext is completely different from the plaintext (i.e. good encryption). If the correlation coefficient equals minus one that means the ciphertext is the negative of the plaintext. So, success of the encryption process means smaller values of the correlation coefficient. so, we are show the different messages in the correlation. The experimental results, the correlation coefficient value of the proposed encryption algorithm shows in the table 5.4.

**Table 3 :**

| The Correlation test from plaintext to ciphertext | |
|---|---|
| Algorithms | Correlation |
| LT : message1 | 0.299632 |
| LT : message2 | 0.554083 |
| RSA algorithm: message1 | 0.14377 |
| RSA algorithm: message2 | 0.031929 |
| RSALT: message1 | 0.00137 |
| RSALT: message2 | 0.656713 |

### E. Comparison between LT, RSA and LTRSA:

In the below table 4 represents comparison between algorithm application of Laplace Transform, RSA algorithm and RSA algorithm with Laplace Transform cryptosystem.

**Table 4.**

| Parameters | Application of Laplace Transform | RSA algorithm | RSA algorithm with LT |
|---|---|---|---|
| Algorithm | Symmetric | Asymmetric | Asymmetric |
| Number of keys | finite series f(t) and mod n | Public key: {e, n} Private key:{d, n} | Public key: { e, n} Private key: {f(t), d, n} |
| Public key size Effect | If size increases execution time also increases | If size increases than time for key generation and execution time also increases and it takes less than LT | If size increases than time for key generation and execution time also increases and it takes more than RSA algorithm |
| Execution time | More than RSA and RSALT | Less than LT and RSALT | Less than LT and More than RSA |
| Speed | Slower than RSA algorithm | Faster than RSALT | Faster than LT and Slower RSA |
| Security | Secure if number of multiples of mod n | High secure if takes large prime numbers | High secure if takes large prime numbers and finite series |
| Complexity | Not so complex easy step of working | So complex for find two large prime numbers | It has complex for finding private key |
| Advantage | Reputation not allowed | Reputation allowed | Reputation not allowed |

## V. CONCLUSION AND FUTURE SCOPE

Recent Application of Laplace Transform for cryptography introduced by Nagalakshmi et al[3] and Hiwerakar [4] are based on the modular arithmetic. In Cryptanalysis by Gupta & Mishra [5], they are using Laplace transform of some elementary modular arithmetic without knowing the secret key. They decrypt the information and moving their internet services easily.

The proposed work expands on innovative method using RSA algorithm with application of Laplace Transform of function providing p & q of two large prime numbers. We can implement high security compared with in this field of other work also. Our main concept is it is impossible to break the algorithm without knowing private key. In future the work can be extended for Sumudu Transform in Cryptography, Integral Transform in cryptography, Laplace - Mellin Transform in Cryptography, El-Zaki Transform in Cryptography, MAHGOUB Transform in Cryptography and so on.

## REFERENCES


1. Grewal BS (2005) *Higher Engineering Mathematics*. Khanna Pub, Delhi.
2. Behrouz A. Forouzan, Debdeep Mukhopadhyay (2010) Cryptography and network security, 2nd edition, McGraw-HillCompanies.
3. Nagalakshmi G, Ravi Kumar B, Chandra sekhar (2011) *Cryptographic scheme of Laplace Transforms*. International Journal of Mathematical Archive-2(12), Vol 2, No12 page: 2515-2519. ISSN 2229-5046 https://www.ijma.info
4. Hiwarekar A P (2013) *Application of Laplace Transforms for Cryptographic scheme*. Proceeding of the world congress on Engineering Vol I, WCE 2013 July 3-5 London U.K. ISSN:2078-0966 www.iaeng.org/publication/WCE2013/WCE2013_pp 95-100.pdf
5. Gupta P ,Mishra P R (2014) *Cryptanalysis of A New Method of Cryptography Using Laplace Transform*. Proceedings of the ThirdInternational Conference on Soft Computing for Problem Solving. Advances in Intelligent Systems and Computing, pp.539 546, Springer, New Delhi. www.springer.com DOI:10.1007/978-81-322-1771-8
6. Roberto P , Briones (2018) *Modification of an encryption scheme based on the Laplace Transform* International Journal of Current Research, Vol 10, Issue 07, pp.71759-71763, https://doi.org/10.24941/ijcr.31417.07.2018
7. Sukalya Som, Moumita Som (2012) *DNA Secret Writing With Laplace Transform*, International Journal of Computer Applications (0975-8887) Volume 50 – No.5, pp. 44-50. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.258.9591
8. Shaish Jamir salim, Mundhe Ganesh Ashruji (2016*) Application of El-Zaki Transform in Cryptography*, International Journal of Modern Sciences and Engineering Technology Vol 3, pp.46-48. Available at https://www.researchgate.net/publication/304244853
9. Moharrem Tuncay GENCOGLU (2017) *Cryptanalysis of Application of Laplace Transform for Cryptography*. ITM Web of Conferences vol .13. https://doi.org/10.1051/itmcomf/20171301009
10. Narander Kumar, Priyanka Chowdary (2016) Performance Evalution of Encryption/Decryption Mechanism to Enhance data Security. Indian journal of Science and Technology, Vol 9(20), DOI: 10.17485/ijst/v9i20