IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 18, Issue 6 Ser. I (Nov. – Dec. 2022), PP 56-65
www.iosrjournals.org

SEE Transformation Based Cryptographic Technique to Encrypt and Decrypt Plaintext

[1]Srushti Gandhi, [2]Ravi Gor
[1]Research Scholar, Department of Mathematics, Gujarat University, Gujarat, India
[2]Department of Mathematics, Gujarat University, Gujarat, India

ABSTRACT
In the era of internet, transmitting data through network communication via mail, social group, online banking and many more are essential. Hence the requirement of securing the information occurs. Moreover, cryptographic techniques like symmetric key, asymmetric key and hashing techniques are employed. In this paper, plaintext is encrypted and decrypted using SEE transformation.

KEYWORDS: *Encryption, Decryption, RSA, SEE transformation*

Date of Submission: 10-11-2022 Date of Acceptance: 25-11-2022

I. INTRODUCTION

Encrypting and decrypting the data is part of the cryptographic process. It helps to hide the information or data. Data is transformed into an unreadable form for unauthorized parties. An encryption helps to transform plain text into cipher text under the control of a secret key. The reverse process is done for decryption.

One of the most primitive ciphers is the Caesar cipher or Shift cipher. Each letter in the plaintext is replaced by a letter that is located a specific number of positions further down the alphabet in this form of substitution cipher. For example, if the key was three, then the plain text A would be replaced by the cipher text D, the next letter B would be replaced by $E$and so on. This is the procedure for encrypting a message. This can be mathematically represented as $p(t) \equiv (t+k) mod\ 26$. The function $p$is allocated to the non-negative integer t,$t \le 26$, the integer in the set $\{ 1,2,3, \ldots 26 \}$ with $p(t) \equiv (t+k) mod\ 26$.

In this paper, research model will encrypt and decrypt a message by using a new integral transformation called SEE transformation. SEE transformation is derived from the Laplace transformation and is widely used in applied mathematics and engineering fields. Based on the mathematical simplicity of this transformation and its fundamental properties, encryption and decryption algorithms are applied to get the message in a straightforward and secure way.

II. LITERATURE REVIEW

Swati et.al [1](2016) had proposed a method of cryptography, in which they used Laplace transform for encrypting the plain text and corresponding inverse Laplace transform for decryption. Their algorithm was effortless and robust. The time complexity of the proposed model was comparatively low and offered higher level of security.

Senthil and Vasuki [2](2018) planned an encryption and decryption algorithm for a message by using Mahgoub transformation and congruence modulo operator.

Dharshini et.al.[4](2019) portrayed a procedure to encrypt and decrypt a message by using an integral transform called N - transform and congruence modulo operator. The approach enables the plain text message in double time safety form. Thus, the process of plain text security was strengthened along with simplified decryption procedure.

Eman et.al [3](2021) applied SEE transform to address straight normal differential conditions with consistent coefficients. Also SEE change of incomplete derivative was concluded and its suitability showed utilizing three was conditional. Its suitability showed utilizing: wave equation, heat equation and Laplace equation, they found the solutions of these equations.

Srushti, and Gor [5](2022) utilized a key for image encryption and decryption produced using RSA and Linear Feedback Shift Register (LFSR). They suggested a method which was extremely sensitive to the LFSR's initial state. The first key is generated by RSA, and the second key was generated by LFSR using the first key. Then both keys were XORed together to produce a strong final key. As a result, in terms of guessing that key, the hacker will have a great difficulty.

III. TERMINOLOGIES

SEE TRANSFORMATION[3]

By manipulating the Laplace transformation, Sadiq, Emad, and Eman (SEE) designed the SEE transformation. It was created with the intention of managing common and halfway differential time conditions.

Frequently, Fourier, Laplace, Elzaki, Aboodh, Mohanad, Al-Zughair, Kamal and Mahgoub transformations are the helpful numerical tools for addressing differential conditions. Similarly, SEE transformation and a portion of its crucial properties are utilized to tackle differential conditions.

SEE transformation was characterized for the outstanding capacities in set A characterized by:

$$A = \{f(t)/ \ni \ M, l_1, l_2 > 0;\ |f(t)| < Me^{l_i|t|}, ift \in (-1)^i x[0,\infty) \}$$

Where M is constant with limited number,
l_1, l_2 might be limited or boundless.

SEE transformation $S(.)$ can be defined as:

$$S[f(t)] = T(v) = \frac{1}{v^n}\int_0^\infty f(t)e^{-vt}dt,\ t \geq 0, l_1 \leq v \leq l_2 \qquad \dots\dots(1)$$

The variable v in this vital change is utilized to figure the variable t the contention of the capacity f.

If $S[f(t)] = T(v)$ is the SEE integral transformation, then $[f(t)] = S^{-1}[T(v)]$ is called an inverse of the SEE integral transformation.

Some Standard Functions:

$f(t)$	$S[f(t)]$
1	$\frac{1}{v^{n+1}}$
t	$\frac{1}{v^{n+2}}$
t^m	$\frac{m!}{v^{m+n+1}};\ m,n > 0$
e^{at}	$\frac{1}{v^n(v-a)}$; where a is constant
$Sin(at)$	$\frac{1}{v^n(v^2+a^2)}$
$Cos(at)$	$\frac{1}{v^{n-1}(v^2+a^2)}$
$Sinh(at)$	$\frac{1}{v^n(v^2-a^2)}$
$Cosh(at)$	$\frac{1}{v^{n-1}(v^2-a^2)}$

RSA ALGORITHM:[5]

RSA algorithm is the most popular asymmetric encryption algorithm. Rivets, Shamir, and Adelman initially presented this algorithm publicly. One of the earliest significant developments in public-key cryptography was RSA, the first known technique for combining signature and encryption. It uses a pair of keys, but only one of them is used to encrypt data in such a way that only the other key in the pair can decrypt it.

There is no logical way to generate the keys among them, even though they are generated using a common technique. The security of RSA is grounded on the concept that factoring a large number is difficult. It depends on the prime factors that are used to encrypt and decrypt the data. The function of multiplying two large prime integers is one-way. Integer multiplication to get a product is easy but factoring the result to identify the two large prime numbers that were previously multiplied is very tough. It is known as a factoring problem.

IV. METHODOLOGY

When Bob passes a message to Alice, the message is converted into polynomial by assigning them ASCII values. Then, SEE transformation is applied to that polynomial to get cipher text. Alice receives the message in an unreadable form. To decrypt this cipher message, inverse SEE transformation is applied to convert cipher text into readable message. Now, Alice can read the message.

On the other hand, RSA algorithm is used to make above algorithm more secure. Keys are generated using RSA algorithm: a public key and a private key. Here, public key is used to encrypt the message and private key is used to decrypt the message. When Bob passes a message to Alice, the message is converted into polynomial by assigning them ASCII values. Then, SEE transformation is applied to that polynomial to get intermediate cipher text. Now, to get encrypted message, RSA encryption algorithm is applied. Alice receives this message in an unreadable form. The message is decrypted using the RSA decryption algorithm to obtain the intermediate cipher text. Then inverse SEE transformation is applied to convert cipher text into readable message. Now, Alice can read the message.

V. ALGORITHM:1

ENCRYPTION ALGORITHM:

i. Choose plaintext. Assign ASCII values to plain text message.
ii. The plain text message is ordered as a finite sequence of numbers.
iii. If n is the number of terms in the sequence, then consider a polynomial $p(t)$ of degree $n - 1$.
iv. Apply SEE transform of polynomial $p(t)$.
v. Find $r_i$such that $q_i \equiv r_i mod\ 255$ for each $i, 1 \le i \le n$.
vi. Consider a new finite sequence $r_1, r_2, r_3, \dots r_n$.
The output text message is in cipher text.

DECRYPTION ALGORITHM:

i. Convert the cipher text into finite sequence of numbers $r_1, r_2, r_3, \dots r_n$.
ii. Let $q_i \equiv 255\, c_i + r_i, \forall i = 1,2,3, \dots n$.
iii. Let$p(t)$ be$\sum_{i=1}^{m} \frac{q_i}{v_{i+1}}$.
iv. Take the inverse SEE transform.
v. Take coefficient of a polynomial $p(t)$ as a finite sequence.
vi. Translate the number of the finite sequence into plaintext.
The output is original plaintext.

VI. EXAMPLE

ENCRYPTION ALGORITHM:

i. Choose plaintext. Assign ASCII values to plain text message.

P	l	@	!	n	T	e	x	t
80	108	64	33	110	84	101	120	116

ii. The plain text message is organized as a finite sequence of numbers.
$80, 108, 64, 33, 110, 84, 101, 120, 116$

iii. If n is the number of terms in the sequence, then consider a polynomial $p(t)$ of degree $n - 1$.
Here, $n = 9$. So, the polynomial is of degree 8.
$p(t) = 80 + 108t + 64t^2 + 33t^3 + 110t^4 + 84t^5 + 101t^6 + 120t^7 + 116t^8$

iv. Apply SEE transform of polynomial $p(t)$.

$$S[p(t)] = 80S[t^0] + 108S[t] + 64S[t^2] + 33S[t^3] + 110S[t^4] + 84S[t^5] + 101S[t^6] + 120S[t^7] + 116S[t^8]$$

$$= 80\ \frac{0!}{v^{0+n+1}} + 108\frac{1!}{v^{1+n+1}} + 64\frac{2!}{v^{2+n+1}} + 33\frac{3!}{v^{3+n+1}} + 110\frac{4!}{v^{4+n+1}} + 84\frac{5!}{v^{5+n+1}} + 101\frac{6!}{v^{6+n+1}} + 120\frac{7!}{v^{7+n+1}} + 116\frac{8!}{v^{8+n+1}}$$

$$= 80\ \frac{1}{v^{n+1}} + 108\frac{1}{v^{n+2}} + 64\frac{2}{v^{n+3}} + 33\frac{6}{v^{n+4}} + 110\frac{24}{v^{4+n+1}} + 84\frac{120}{v^{n+6}} + 101\frac{720}{v^{n+7}} + 120\frac{5040}{v^{n+8}} + 116\frac{40320}{v^{n+9}}$$

$$= \frac{80}{v^{n+1}} + \frac{108}{v^{n+2}} + \frac{128}{v^{n+3}} + \frac{198}{v^{n+4}} + \frac{2640}{v^{4+n+1}} + \frac{10080}{v^{n+6}} + \frac{72720}{v^{n+7}} + \frac{604800}{v^{n+8}} + \frac{4677120}{v^{n+9}}$$

v. Find ri such that $q_i \equiv r_i mod\ 255$ for each $i, 1\ \leq i\ \leq n$.
Here,$q_1 = 80,\ q_2 = 108,\ q_3 = 128, q_4 = 198\ , q_5 = 2640,\ q_6 = 10080,\ q_7 = 72720,\ q_8 = 604800,$ $q_9 = 4677120$

q_i	$q_i \equiv r_i mod\ 255$	r_i
80	$80 \equiv\ 80 mod\ 255$	80
108	$108 \equiv\ 108 mod\ 255$	108
128	$128 \equiv\ 128 mod\ 255$	128
198	$198 \equiv\ 198 mod\ 255$	198
2640	$2640 \equiv\ 90 mod\ 255$	90
10080	$10080 \equiv\ 135 mod\ 255$	135
72720	$72720\ \equiv\ 45 mod\ 255$	45
604800	$604800 \equiv\ 195 mod\ 255$	195
4677120	$4677120 \equiv 165 mod\ 255$	165

vi. Consider a new finite sequence $r_1\ , r_2\ , r_3\ \ , \dots\ r_n$.
$80, 108, 128, 198, 90, 135, 45, 195, 165$

vii. The output text message is in cipher text.

80	108	128	198	90	135	45	195	165
P	l	Ç	ã	Z	ç	–	⊦	Ñ

The cipher text is Pl Ç ã Z ç – ⊦Ñ.

DECRYPTION ALGORITHM:

i. Convert the cipher text into finite sequence of numbers $r_1\ , r_2\ , r_3\ \ , \dots\ r_n$.

80	108	128	198	90	135	45	195	165
P	l	Ç	ã	Z	ç	–	⊦	Ñ

ii. Let $q_i \equiv 255\ c_i +\ r_i\ , \forall i = 1,2,3, \dots n$.

r_i	$q_i \equiv 255\ c_i +\ r_i$	q_i
80	$80 \equiv\ 255(0)\ + 80$	80
108	$108 \equiv\ 255(0)\ + 108$	108
128	$128 \equiv\ 255\ (0) +\ 128$	128
198	$198 \equiv\ 255\ (0) + 198$	198
90	$2640 \equiv\ 255(10)\ + 90$	2640
135	$10080 \equiv\ 255\ (39) + 135$	10080
45	$72720\ \equiv\ 255(285)\ + 45$	72720
195	$604800 \equiv\ 255\ (2371) + 195$	604800
165	$4677120 \equiv 255\ (18341) + 165$	4677120

iii. Let $p(t)$ be $\sum_{i=1}^{m} \frac{q_i}{v_{i+1}}$.

$$p(t) = \frac{80}{v^{n+1}} + \frac{108}{v^{n+2}} + \frac{128}{v^{n+3}} + \frac{198}{v^{n+4}} + \frac{2640}{v^{4+n+1}} + \frac{10080}{v^{n+6}} + \frac{72720}{v^{n+7}} + \frac{604800}{v^{n+8}} + \frac{4677120}{v^{n+9}}$$

iv. Take the inverse SEE transform.

$$S^{-1}[p(t)] = S^{-1}\left[\frac{80}{v^{n+1}}\right] + S^{-1}\left[\frac{108}{v^{n+2}}\right] + S^{-1}\left[\frac{128}{v^{n+3}}\right] + S^{-1}\left[\frac{198}{v^{n+4}}\right] + S^{-1}\left[\frac{2640}{v^{4+n+1}}\right] + S^{-1}\left[\frac{10080}{v^{n+6}}\right] + S^{-1}\left[\frac{72720}{v^{n+7}}\right] + S^{-1}\left[\frac{604800}{v^{n+8}}\right] + S^{-1}\left[\frac{4677120}{v^{n+9}}\right]$$

$$= 80\ \frac{0!}{v^{0+n+1}} + 108\frac{1!}{v^{1+n+1}} + 64\frac{2!}{v^{2+n+1}} + 33\frac{3!}{v^{3+n+1}} + 110\frac{4!}{v^{4+n+1}} +\ 84\frac{5!}{v^{5+n+1}} + 101\frac{6!}{v^{6+n+1}} + 120\frac{7!}{v^{7+n+1}} + 116\frac{8!}{v^{8+n+1}}$$

$$= 80S[t^0] + 108S[t] + 64S[t^2] + 33S[t^3] + 110S[t^4] + 84S[t^5] + 101S[t^6] + 120S[t^7] + 116S[t^8]$$

$$= 80 + 108t + 64t^2 + 33t^3 + 110t^4 + 84t^5 + 101t^6 + 120t^7 + 116t^8$$

v. The coefficient of a polynomial $p(t)$ as a finite sequence.
$80, 108, 64, 33, 110, 84, 101, 120, 116$

vi. Translate the number of the finite sequence into plaintext.

80	108	64	33	110	84	101	120	116
P	l	@	!	n	T	e	x	t

vii. The output is original plaintext.
Pl@!nText

VII. ALGORITHM: 2

Furthermore, RSA is used to make this algorithm more secure.

KEY GENERATION:[5]
Generate a key using RSA algorithm.
With suitably long passwords, it is usually assumed that RSA is safe. Find two prime numbers and use those two prime numbers to generate a pair of keys.
Step 1: First choose the two distinct prime numbers p and q.
For security purposes, the integer p and q should be chosen similar to bit-length. Prime integers can be efficiently found by a primality testing.

Step 2: Compute the n value, $n = pq$.
n is used as the modulus for both the public and private keys. Its length, usually expressed in bits, is the key length.

Step 3: Compute $\varphi(n) = \varphi(p)\varphi(q) = (p - 1)(q - 1) = n - (p + q - 1)$,
where φ is Euler's totient function. This value is kept private.

Step 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.
Here, e is public key. e has a short bit-length. this weight results in more efficient encryption. However, much smaller values of e have been shown to be less secure in some settings.

Step 5: Determine $d$as $d \equiv e - 1\ (mod\varphi(n))$; i.e., d is the modular multiplicative inverse of $e(modulo\varphi(n))$. This is stated as, solve the d given $d \cdot e \equiv 1\ (mod\varphi(n))$. This is computed using extended Euclidean algorithm. It is using the pseudo code in the Modular integers section, inputs a and n correspond to e and $\varphi(n)$, respectively.

Step 6: d value is kept as the private key. The public key consists of the modulus$n$ and the public key e.
The private key has the modulus n and the private key d, and it keep in secret. $p, q,$ and $\varphi(n)$ values are kept in secret because they can be used to calculate d.

The generated sequence is intern converted into a sequence of 8-bit binary and which is used in the generation of key sequences.

ENCRYPTION ALGORITHM:
i. Choose plaintext. Assign ASCII values to plain text message.
ii. The plain text message is organized as a finite sequence of numbers.
iii. If n is the number of terms in the sequence, then consider a polynomial $p(t)$ of degree $n - 1$.
iv. Apply SEE transform of polynomial $p(t)$.
v. Find ri such that $q_i \equiv r_i mod\ 255$ for each $i, 1 \leq i \leq n$.
vi. Consider a new finite sequence $r_1, r_2, r_3, \ldots r_n$.
The output text message is in intermediate cipher text.

Encryption by RSA:[5]
Step:1 Alice transmits public key to Bob, keeping her private key secret.
Step: 2 When Bob wishes to send the message "P" to Alice, he first converts P to an integer

such that $0 < m < n$ by using agreed upon reversible protocol known as a padding scheme.
Step:3 Bob computes, with Alice's public key information, the ciphertext c corresponding to

$$c \equiv m^e \ (mod n).$$

Bob now sends message "P" in ciphertext, or c, to Alice.
The output plaintext is ciphertext.

DECRYPTION ALGORITHM:[5]
Decryption by RSA:
Step:1 Alice recovers P from c by using his private key exponent, d, by the computation

$$m \equiv c^d \ (mod n).$$

Step:2 Given P, Alice can recover the message "P" by reversing the padding scheme.
The output is intermediate cipher text.
Now,

i. Convert the cipher text into finite sequence of numbers $r_1, r_2, r_3, \dots r_n$.
ii. Let $q_i \equiv 255\, c_i + r_i, \forall i = 1,2,3, \dots n$.
iii. Let $p(t)$ be $\sum_{i=1}^{m} \frac{q_i}{v_{i+1}}$.
iv. Take the inverse SEE transform.
v. The coefficient of a polynomial $p(t)$ as a finite sequence.
vi. Translate the number of the finite sequence into plaintext.

The output is original plaintext and Alice can now read the message.

VIII. EXAMPLE

KEY GENERATION:
By using RSA algorithm,
Step: 1 $P = 227, Q = 313;$

Variable	Value	Name	Formula	Description
N	71051	Modulus	$N = P * Q$	Product of 2 prime numbers
L	70512	Length	L: (p - 1) * (q - 1)	Another way of calculating 'L' is to list of numbers from 1 to N, remove numbers which have common factor which N and count the remaining numbers.
E	5	Encryption key		Find a number between 1 and L that is coprime with L and N.
D	98717	Decryption key	D * E mod L = 1	Remainder of the product of D and E when divided by L should be 1 (D * E % L = 1)

∴ Public key $(E, N) = (5,71051)$
Private key $(D, N) = (98717,71051)$

ENCRYPTION ALGORITHM:

i. Choose plaintext. Assign ASCII values to plain text message.

P	l	@	!	n	T	e	x	t
80	108	64	33	110	84	101	120	116

ii. The plain text message is organized as a finite sequence of numbers.
$80, 108, 64, 33, 110, 84, 101, 120, 116$

iii. If n is the number of terms in the sequence, then consider a polynomial $p(t)$ of degree $n - 1$.
Here, $n = 9$. So, the polynomial is of degree 8.
$p(t) = 80 + 108t + 64t^2 + 33t^3 + 110t^4 + 84t^5 + 101t^6 + 120t^7 + 116t^8$

iv. Apply SEE transform of polynomial $p(t)$.

$$S[p(t)] = 80S[t^0] + 108S[t] + 64S[t^2] + 33S[t^3] + 110S[t^4] + 84S[t^5] + 101S[t^6] + 120S[t^7] + 116S[t^8]$$

$$= 80\ \frac{0!}{v^{0+n+1}} + 108\frac{1!}{v^{1+n+1}} + 64\frac{2!}{v^{2+n+1}} + 33\frac{3!}{v^{3+n+1}} + 110\frac{4!}{v^{4+n+1}} + 84\frac{5!}{v^{5+n+1}} + 101\frac{6!}{v^{6+n+1}} + 120\frac{7!}{v^{7+n+1}} + 116\frac{8!}{v^{8+n+1}}$$

$$= 80\ \frac{1}{v^{n+1}} + 108\frac{1}{v^{n+2}} + 64\frac{2}{v^{n+3}} + 33\frac{6}{v^{n+4}} + 110\frac{24}{v^{4+n+1}} + 84\frac{120}{v^{n+6}} + 101\frac{720}{v^{n+7}} + 120\frac{5040}{v^{n+8}} + 116\frac{40320}{v^{n+9}}$$

$$= \frac{80}{v^{n+1}} + \frac{108}{v^{n+2}} + \frac{128}{v^{n+3}} + \frac{198}{v^{n+4}} + \frac{2640}{v^{4+n+1}} + \frac{10080}{v^{n+6}} + \frac{72720}{v^{n+7}} + \frac{604800}{v^{n+8}} + \frac{4677120}{v^{n+9}}$$

v. Find ri such that $q_i \equiv r_i mod\ 255$ for each $i, 1\ \leq i\ \leq n$.
Here,$q_1 = 80,\ q_2 = 108,\ q_3 = 128, q_4 = 198\ , q_5 = 2640,\ q_6 = 10080,\ q_7 = 72720,\ q_8 = 604800,$ $q_9 = 4677120$

q_i	$q_i \equiv r_i mod\ 255$	r_i
80	$80 \equiv\ 80 mod\ 255$	80
108	$108 \equiv\ 108 mod\ 255$	108
128	$128 \equiv\ 128 mod\ 255$	128
198	$198 \equiv\ 198 mod\ 255$	198
2640	$2640 \equiv\ 90 mod\ 255$	90
10080	$10080 \equiv\ 135 mod\ 255$	135
72720	$72720\ \equiv\ 45 mod\ 255$	45
604800	$604800 \equiv\ 195 mod\ 255$	195
4677120	$4677120 \equiv 165 mod\ 255$	165

vi. Consider a new finite sequence $r_1\ , r_2\ , r_3\ \ , \dots\ r_n$.
$80, 108, 128, 198, 90, 135, 45, 195, 165$

vii. The output text message is in cipher text.

80	108	128	198	90	135	45	195	165
P	l	Ç	ã	Z	ç	$-$	⊦	Ñ

The intermediate cipher text is Pl Ç ã Z ç $-$ ⊦Ñ.

Now, by using RSA algorithm,
To encrypt a message, message $\hat{}E\ \%\ N$ is applied to intermediate cipher.
Message: Pl Ç ã Z ç $-$ ⊦Ñ
Encrypted Message: 69982,5019,20067,70370,64543,28037,8678,69677,6336

DECRYPTION ALGORITHM:
Now, by using RSA algorithm,
To decrypt a message, $encrypted\_message\hat{}D\ \%\ N$ is applied to encrypted cipher.
Cipher message: 69982,5019,20067,70370,64543,28037,8678,69677,6336
Message decrypted to ASCII code: 80,108,199,227,90,231,45,9500,209
Intermediate cipher Message: PlÇãZç $-$ ⊦Ñ

i. Convert the cipher text into finite sequence of numbers $r_1\ , r_2\ , r_3\ \ , \dots\ r_n$.

80	108	128	198	90	135	45	195	165
P	l	Ç	ã	Z	ç	$-$	⊦	Ñ

ii. Let $q_i \equiv 255\ c_i +\ r_i\ , \forall i = 1,2,3, \dots n$.

r_i	$q_i \equiv 255\ c_i +\ r_i$	q_i
80	$80 \equiv\ 255(0)\ + 80$	80
108	$108 \equiv\ 255(0)\ + 108$	108
128	$128 \equiv\ 255\ (0) +\ 128$	128
198	$198 \equiv\ 255\ (0) + 198$	198
90	$2640 \equiv\ 255(10)\ + 90$	2640
135	$10080 \equiv\ 255\ (39) + 135$	10080

45	$72720 \equiv 255(285) + 45$	72720
195	$604800 \equiv 255\,(2371) + 195$	604800
165	$4677120 \equiv 255\,(18341) + 165$	4677120

iii. Let $p(t)$ be $\sum_{i=1}^{m}\frac{q_i}{v_{i+1}}$.

$$p(t) = \frac{80}{v^{n+1}} + \frac{108}{v^{n+2}} + \frac{128}{v^{n+3}} + \frac{198}{v^{n+4}} + \frac{2640}{v^{4+n+1}} + \frac{10080}{v^{n+6}} + \frac{72720}{v^{n+7}} + \frac{604800}{v^{n+8}} + \frac{4677120}{v^{n+9}}$$

iv. Take the inverse SEE transform.

$$S^{-1}[p(t)] = S^{-1}\left[\frac{80}{v^{n+1}}\right] + S^{-1}\left[\frac{108}{v^{n+2}}\right] + S^{-1}\left[\frac{128}{v^{n+3}}\right] + S^{-1}\left[\frac{198}{v^{n+4}}\right] + S^{-1}\left[\frac{2640}{v^{4+n+1}}\right] + S^{-1}\left[\frac{10080}{v^{n+6}}\right] + S^{-1}\left[\frac{72720}{v^{n+7}}\right] + S^{-1}\left[\frac{604800}{v^{n+8}}\right] + S^{-1}\left[\frac{4677120}{v^{n+9}}\right]$$

$$= 80\,\frac{0!}{v^{0+n+1}} + 108\,\frac{1!}{v^{1+n+1}} + 64\,\frac{2!}{v^{2+n+1}} + 33\,\frac{3!}{v^{3+n+1}} + 110\,\frac{4!}{v^{4+n+1}} + 84\,\frac{5!}{v^{5+n+1}} + 101\,\frac{6!}{v^{6+n+1}} + 120\,\frac{7!}{v^{7+n+1}} + 116\,\frac{8!}{v^{8+n+1}}$$

$$= 80S[t^0] + 108S[t] + 64S[t^2] + 33S[t^3] + 110S[t^4] + 84S[t^5] + 101S[t^6] + 120S[t^7] + 116S[t^8]$$

$$= 80 + 108t + 64t^2 + 33t^3 + 110t^4 + 84t^5 + 101t^6 + 120t^7 + 116t^8$$

v. The coefficient of a polynomial $p(t)$ as a finite sequence.
$80, 108, 64, 33, 110, 84, 101, 120, 116$

vi. Translate the number of the finite sequence into plaintext.

80	108	64	33	110	84	101	120	116
P	l	@	!	n	T	e	x	t

vii. The output is original plaintext.
Pl@!nText

IX. ANALYSIS

CORRELATION:

Correlation is used to measure the degree of the relationship between linearly related variables. Although interpretations of relationship strength or effect size differ between variables, the table below provides general rules:

Correlation Coefficient	Strength	Direction
Greater than 0.5	Strong	Positive
Between 0.3 to 0.5	Moderate	Positive
Between 0 to 0.3	Weak	Positive
0	None	None
Between 0 to -0.3	Weak	Negative
Between -0.3 to -0.5	Moderate	Negative
Less than -0.5	Strong	Negative

Here, the original text and cipher text in algorithm1 is 21.9% negatively correlated. Thus, the original message and cipher text are weakly correlated to each other. In other words, original message and cipher text are different form each other.

The original text and cipher text in algorithm2 is 27.9% negatively correlated. Thus, the original message and cipher text are weakly correlated to each other. In other words, original message and cipher text are completely different form each other.

MEAN SQUARE ERROR:

The mean square error (MSE) is defined as mean or average of the square of the difference between actual and estimated values. It measures the number of errors. It assesses the average squared difference between the

observed and predicted values. When a model has no error, the MSE equals zero. As model error increases, its value increases.
MSE can be calculated by:

$$\text{MSE} = \frac{\Sigma(y_i - \widehat{y_i})^2}{n}$$

Where:
y_i is the i^{th}observed value.
$\widehat{y_i}$ is the corresponding predicted value.
n is the number of observations.

Here, the MSE of algorithm1 is 5053.777777777777 and that of algorithm2 is 329040355.0.
Hence, as MSE is higher in algorithm2, encrypted and original plaintext are completely different.

ROOTED MEAN SQUARE ERROR:

Root Mean Squared Error (RMSE) is the square root of the mean squared error (MSE) between the predicted and actual values. It represents that the concentrated data is around the line of best fit or not. Root mean square error is commonly used to verify experimental results.

RMSE can be calculated by:

$$\text{RMSE} = \sqrt{\frac{\Sigma(y_i - \widehat{y_i})^2}{n}}$$

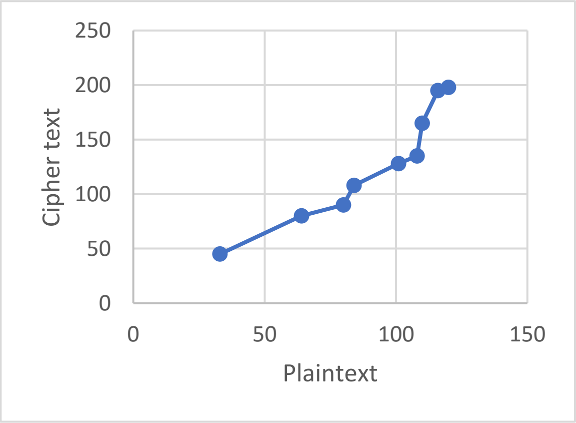
Here, the RMSE of algorithm1 is 71.089921 algorithm2 is 18139.46953469147.

X. RESULTS AND DISCUSSION

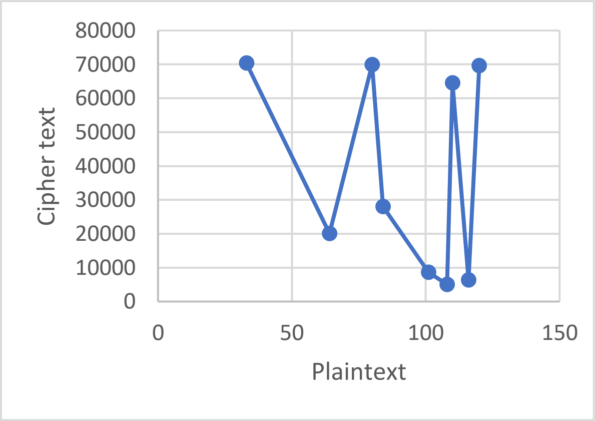
Original plaintext = 80, 108, 64, 33, 110, 84, 101, 120, 116
For algorithm1:
Cipher text = 80, 108, 128, 198, 90, 135, 45, 195, 165
For algorithm2:
Public key = 5
Private Key = 98717
Intermediate cipher text = 80, 108, 128, 198, 90, 135, 45, 195, 165
Cipher text = 69982,5019,20067,70370,64543,28037,8678,69677,6336

Graphs below are a diagram where each value in the data set is represented by a dot. It shows a relationship between plaintext and cipher text. Graph shows that algorithm2 has much more variations between cipher text and plaintext as compared to algorithm1.

For Algorithm1



For Algorithm2



The proposed algorithms are simple, straight forward but essentially strong and compact approach to cryptography. It provides the same or sometimes even better level of security using minimal time complexity.

The key generation scheme, which is entirely based on the input given in algorithm2, can be used for a fraud prevention mechanism. Proposed algorithms provide as many transformations as per the requirements which are the most useful factor for changing key. Keys can be generated through various algorithms like RSA, DES, El Gamal, etc. Therefore, it is exceedingly difficult for an eyedropper to trace the key by any attack.

XI. CONCLUSION

In this proposed work, a new integral transform SEE transform with congruence modulo operator is used. In the other algorithm, RSA algorithm is used with a new integral transform SEE transform with congruence modulo operator. The results are verified with the capabilities of Python and excel for both the algorithms. RSA algorithm calculations are done with the help of Public Key Cryptography using RSA algorithm calculator. This procedure allows the plain text message in double time safety form. Thus, the process of plain text security is strengthened as well as the process of decryption is simplified.

REFERENCE

[1]. Dhingra, S., Savalgi, A. A., & Jain, S. (2016). Laplace transformation based cryptographic technique in network security. International Journal of Computer Applications, 136(7), 0975-8887.

[2]. Kumar, P. S., & Vasuki, S. (2018). An Application of MAHGOUB Transform in Cryptography. Advances in Theoretical and Applied Mathematics, 13(2), 91-99.

[3]. Mansour, E. A., Kuffi, E. A., & Mehdi, S. A. (2021). The new integral transform “SEE transform” and its applications. Periodicals of Engineering and Natural sciences (PEN), 9(2), 1016-1029.

[4]. Priya, D. S., Amirtha, M. A., & Kumar, S. P. (2019). Application of N-transform in cryptography. International Journal of Scientific Research and Reviews, 8(1), 2143-2147.

[5]. Srushti, G., and Gor, R. (2022). Digital Image Encryption using RSA and LFSR. International Journal of Engineering Science Technologies, 6(4), 1-16. doi: 10.29121/IJOEST.v6.i4.2022.351.

[6]. https://theasciicode.com.ar/

[7]. https://umaranis.com/rsa_calculator_demo.html