Malaya Journal of Matematik, Vol. 8, No. 3, 1203-1205, 2020

https://doi.org/10.26637/MJM0803/0082

A new (k,n) secret sharing symmetric-key cryptographic method using ASCII conversion and Laplace transforms

Binoy Joseph[1]* and Bindhu K. Thomas[2]

Abstract

In this paper we proposed a new encryption algorithm for secret sharing, in which we have used Laplace transform and Lagrangian interpolation for encryption and decryption. We also use a revised form of (K,n) threshold scheme for ensure security.

Keywords

Cryptography, Encryption, Decryption, Plain Text, Cipher Text, Key, Laplace Transform.

AMS Subject Classification

44A12, 68P30, 94A60, 94A62.

[1,2] *Mathematics Research Center, Mary Matha Arts and Science College Mananthavady-670645, affiliated to Kannur University, Kerala, India.*

***Corresponding author**: [1] binoy.sib@gmail.com; [2]bindhukthomas@gmail.com

Article History: Received **12** April **2020**; Accepted **22** July **2020**

Contents

1 Introduction 1203

2 Related Works 1203

3 Proposed Algorithm 1204

3.1 Illustration 1205

3.2 Remark 1205

4 Conclusion 1205

References 1205

1. Introduction

Cryptography is an art of protect information by transforming it to unreadable format called Cipher text. The process of converting plain text to cipher text is called encryption, and the process of converting cipher text to original plain text is called decryption. Cryptographic algorithms are classified as Symmetric-key cryptography and publickey (Asymmetric) cryptography. In Symmetric-key cryptography, each sender and receiver shared the same key to encrypt and decrypt data.

In this paper we proposed a new cryptographic scheme for secret sharing using Laplace Transform.Here we convert the given plain text of size k into a polynomial of degree k using ASCII code and Laplace transform. Using the concept of (k,n)threshold secret sharing scheme we encode the given secret data into n shares and distribute them to n participants. We use inverse Laplace transform and Lagrangian Interpolation method for decryption.

2. Related Works

Encrypting and Decrypting is carried forward using functional mathematical algorithms and secret keys in Trending cryptography.

Adi Shamir [3,4] (1979) proposed first the concept of (k,n) threshold secret sharing scheme which is designed to encode a secret data set into n shares and distribute them to n participants, where any k or more of the shares can be collected to recover the secret data, but any $k-1$ or fewer of them will gain no information about it.After the scheme was proposed, many related topics have been studied (Sun and Shieh, 1994; Chang and Lee, 1993).

Mathematical techniques using matrices for encryption and decryption are found by Dhanorkar and Hiwarekar, Overbey, Traves and Wojdylo, Saeednia. Extensive work is also shown by Sachin and Bani in 2013 and by Swati Dhingra, Archana A.Savalgi and Swati Jain in 2016 by presenting new scheme for the cryptographic purpose by combining infinite series and Laplace transform using ASCII code. The process of encryption is further expanded using series expansion of $f(t)$ and its Laplace transform. The Laplace transform is arranged in the form of an array with keys at even position, While the Decryption is done by inverse Laplace transforms

[1,2]. We use ASCII code along with exponential function in this paper.

"If $f(t)$ is a function defined for all positive values of t, then the *Laplace Transform*[5] of $f(t)$ is defined as

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

provided that the integral exists. Here the parameter s is a real or complex number. The corresponding *inverse Laplace transform* is

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

If

$$L\{f_1(t)\} = F_1(s), L\{f_2(t)\} = F_2(s), ..., L\{f_n(t)\} = F_n(s)$$

then

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

where $c_1, c_2, ... c_n$ are constants" and

$$L\{t^n\} = \tfrac{n!}{s^{n+1}}$$

3. Proposed Algorithm

Select the message, M, of length k to be sent. Convert M into ASCII code as G_0, G_1, G_2,...,G_{k-1}. Choose an arbitrary natural number r as first key.
Define

$$f(t) = \sum_{i=0}^{\infty} G_i \frac{(rt)^i}{i!} \tag{3.1}$$

Here $G_i = 0$ for all $i > k-1$. In the next step we take the Laplace Transform of $f(t)$ and form a polynomial

$$g(s) = s^k L[f(t)] = s^k L[\sum_{i=0}^{k-1} G_i \frac{(rt)^i}{i!}] = \sum_{i=0}^{k-1} G_i (r^i) s^{k-(i+1)} \tag{3.2}$$

In the next step we find n $(n > k-1)$ keys $Q_0, Q_1, Q_2, ..., Q_{n-1}$ in such a way that

$$g(i) = Q_i * 256 + r_i \tag{3.3}$$

where $i = 0, 1, 2, ... n-1$
Now the given plain text gets converted into n shares such as

$$(0, r_0), (1, r_1), (2, r_2), ... (n-1, r_{n-1}) \tag{3.4}$$

The above n shares are then distributed into n participants.
In the decryption stage we use any $m(k \leq m \leq n)$ shares among the n distributed shares as follows
(j, r_j, Q_j) where $j \in 0, 1, 2, ..., (n-1)$

Choose $s_i = j$ where $i = 0, 1, 2, 3, ..., k-1$ and
$f_i = r_j$ where $i = 0, 1, 2, 3, ..., k-1$
In the next step using Lagrangian Interpolation formula

$$f(t) = L^{-1}[\frac{1}{s^k} \sum_{i=0}^{k-1} \frac{(256 * Q_i + f_i) l_i(s)}{l_i(s_i)}] \tag{3.5}$$

which is of the form

$$f(t) = \sum_{i=0}^{k-1} G_i t^i \tag{3.6}$$

Now the plain text can be extract using

$$S_i = \frac{G_i * i!}{r^i} \tag{3.7}$$

where $i = 0, 1, 2, ..., k-1$.

Theorem 3.1. *The given plain text of length k in terms of ASCII codes G_i where $i = 0, 1, 2, ..., k-1$ under Laplace transform of $f(t) = \sum_{i=0}^{k-1} G_i \frac{(rt)^i}{i!}$ can be converted into a cipher polynomial of the form $g(s) = s^k L[f(t)] = \sum_{i=0}^{k-1} G_i' s^{k-(i+1)}$ where $G_i' = G_i r^i$*

Proof. Let given plain text $f(t) = Ge^{rt} = \sum_{i=0}^{k-1} G_i \frac{(rt)^i}{i!}$
now $g(s) = s^k L[f(t)]$
$= s^k L[\sum_{i=0}^{k-1} G_i \frac{(rt)^i}{i!}]$
$= s^k \sum_{i=0}^{k-1} G_i \frac{(r)^i}{i!} L[t^i]$
$= s^k \sum_{i=0}^{k-1} G_i \frac{(r)^i}{i!} \frac{i!}{s^{(i+1)}}$
$= \sum_{i=0}^{k-1} G_i' s^{k-(i+1)}$ where $G_i' = G_i r^i$ □

Theorem 3.2. *The given cipher shares of secret triplet (s_i, f_i, Q_i) for $i = 0, 1, 2, ..., k-1$ with a given key r, can be converted into plain text G_i by multiplying the coefficients by $\frac{i!}{r^i}$ of the polynomial under the inverse Laplace transform of*
$[\frac{1}{s^n} \sum_{i=0}^{k-1} \frac{(256 * Q_i + f_i) l_i(s)}{l_i(s_i)}]$
where $l_i(s) = (s - s_0)(s - s_1)...(s - s_{i-1})(s - s_{i+1})...(s - s_{k-1})$
and
$l_i(s_i) = (s_i - s_0)(s_i - s_1)...(s_i - s_{i-1})(s_i - s_{i+1})...(s_i - s_{k-1})$

Proof. Here $(256 * Q_i + f_i)$ representing the value of required polynomial $g(s)$ corresponding to s_i ,$g(s_i) = 256 * Q_i + f_i$
By Lagrangian interpolation method $\sum_{i=0}^{n-1} \frac{(256 * Q_i + f_i) l_i(s)}{l_i(s_i)}$ represent the polynomial $g(s)$ of degree $k-1$ such that $g(s_i) = 256 * Q_i + f_i$ for $i = 0, 1, 2, ..., k-1$
say $g(s) = \sum_{i=0}^{k-1} G_i' s^i$
Now $\frac{1}{s^k} g(s) = \sum_{i=0}^{k-1} \frac{G_i'}{s^{k-i}} = \frac{G_0'}{s^k} + \frac{G_1'}{s^{k-1}} + \frac{G_2'}{s^{k-2}} + ... + \frac{G_{k-2}'}{s^2} + \frac{G_{k-1}'}{s^1}$
Now $L^{-1}[\frac{1}{s^k} g(s)] = \sum_{i=0}^{k-1} \frac{G_{k-(i+1)}' t^i}{i!}$
$= G_{k-1}' + G_{k-2}' \frac{t}{1!} + G_{k-3}' \frac{t^2}{2!} + ... + G_1' \frac{t^{k-2}}{(k-2)!} + G_0' \frac{t^{k-1}}{(k-1)!}$

$= G_0^{''} + G_1^{''}t + G_2^{''}t^2 + ... + G_{k-1}^{''}t^{k-1}$ where $G_i^{''} = \frac{G_{k-(i+1)}^{'}}{i!}$ for $i = 0,1,2,...,(k-1)$
Now $G_i = G_i^{''}\frac{i!}{r^i}$ $i = 0,1,2,...,(k-1)$ are the secret. □

3.1 Illustration

Let M is "SECRET" be the message to be sent. Here length $k = 6$. The corresponding ASCII code of given plain text is
$G_0 = 83$ $G_1 = 69$ $G_2 = 67$ $G_3 = 82$ $G_4 = 69$ $G_5 = 84$
Take $r = 2$ as the first key. Now using equation (1)
$f(t) = \sum_{i=0}^{5} G_i \frac{(rt)^i}{i!}$
. $= 83 + 69(2t) + 67\frac{(2t)^2}{2!} + 82\frac{(2t)^3}{3!} + 69\frac{(2t)^4}{4!} + 84\frac{(2t)^5}{5!}$
Now from equation (2)
$g(s) = s^6L[f(t)] = 2688 + 1104s + 656s^2 268s^3 + 138s^4 + 83s^5$
Now using equation (3) developing Qi and r_i we have
$g(0) = 2688 = 10 * 256 + 128$
$g(1) = 4937 = 19 * 256 + 73$
$g(2) = 14528 = 56 * 256 + 192$
$g(3) = 50487 = 197 * 256 + 55$
$g(4) = 155072 = 605 * 256 + 192$
$g(5) = 403733 = 1577 * 256 + 21$
$g(6) = 915072 = 3574 * 256 + 128$
$g(7) = 1860803 = 7268 * 256 + 195$
$g(8) = 3475712 = 13577 * 256 + 0$
$g(9) = 6067617 = 23701 * 256 + 161$
$g(10) = 10027328 = 39169 * 256 + 64$
here the shares are $(i, r(i))$ where $i = 0,1,2,...10$. Here the keys are $Q_0, Q_1, Q_2, ..., Q_{10}$ We now distributed the shares among $n = 11$ participants.
In the decryption stage we accept any 6 share and corresponding keys Q_i where $i \in 0,1,2,...,10$. Suppose we accept the shares $(0,128,10), (3,55,197), (5,21,1577), (8,0,13577)$, $(10,64,39169)$ and $(2,192,56)$ of the form (s_i, f_i, Q_i) where $i \in 0,1,2,3,4,5$
$s_0 = 0, s_1 = 3, s_2 = 5, s_3 = 8, s_4 = 10$ and $s_5 = 2$
$f_0 = 128, f_1 = 55, f_2 = 21, f_3 = 0, f_4 = 64$ and $f_5 = 192$
$Q_0 = 10, Q_1 = 197, Q_2 = 1577, Q_3 = 13577, Q_4 = 39169$ and $Q_5 = 56$
Using (5) $f(t) = L^{-1}[\frac{1}{s^n}\sum_{i=0}^{n-1}\frac{(256*Q_i+f_i)l_i(s)}{l_i(s_i)}]$
$l_0(s) = s^5 - 28s^4 + 291s^3 - 1388s^2 + 3020s - 2400$
$l_1(s) = s^5 - 25s^4 + 216s^3 - 740s^2 + 800s$
$l_2(s) = s^5 - 23s^4 + 176s^3 - 508s^2 + 480s$
$l_3(s) = s^5 - 20s^4 + 131s^3 - 340s^2 + 300s$
$l_4(s) = s^5 - 18s^4 + 111s^3 - 278s^2 + 240s$
$l_5(s) = s^5 - 26s^4 + 239s^3 - 910s^2 + 1200s$
$l_0(0) = -2400, l_1(3) = -210, l_2(5) = 450, l_3(8) = -1440$, $l_4(10) = 5600$ and $l_5(2) = 288$
$f(t) = L^{-1}\{\frac{1}{s^6}[83s^5 + 138s^4 + 268s^3 + 656s^2 + 1104s + 2688]\}$
$f(t) = 83 + 138t + 268\frac{t^2}{2!} + 656\frac{t^3}{3!} + 1104\frac{t^4}{4!} + 2688\frac{t^5}{5!}$
Using (7) $S_0 = 83, S_1 = 69, S_2 = 67, S_3 = 82, S_4 = 69$ and $S_5 = 84$
The decrypted message is "SECRET"

3.2 Remark

For different values of r we get different shares for decryption. In the above example when $r = 1$ shares and corresponding keys are
$(0,84,0), (3,180,111), (5,10,1224), (8,252,11884)$, $(10,246,35413)$ and $(2,238,18)$
when $r = 3$ shares and corresponding keys are
$(0,188,79), (3,242,430), (5,24,2218), (8,228,15949)$, $(10,46,44026)$ and $(2,38,200)$
when $r = 4$ shares and corresponding keys are
$(0,0,336), (3,173,1006), (5,243,3403), (8,0,19384)$, $(10,160,50466)$ and $(2,32,617)$

4. Conclusion

In the proposed work we expand an innovative cryptographic scheme for sharing a secret using Laplace transform of exponential function using ASCII code. To reduce the crypt -analysis attack risk, a powerful key theory plays vital role. The two types of keys makes the scheme more secure. Some authors uses even degree expansions (Eg.$L(\cosh rt)$) for taking Laplace Transforms which gives the second level keys are more lengthy. In our proposed algorithm we use exponential function which gives keys are small as compared to other algorithms. By changing the first key, r the algorithm provides many transformations for the same secret.

References

[1] A.P. Hiwarekar, Application of Laplace Transform for Cryptography, *IJESR*, 5(2015), 129–135.

[2] C.H. Jayanthi and V. Srinivas, Mathematical Modelling for Cryptography using Laplace Transform. *IJMTT*, 65(2019)(2), 10–15.

[3] A. Shamir, How to share a secret, *Communications of the Association for Computing Machinery*, 22 (1979), 612–613.

[4] G. R. Blakely, Safeguarding cryptographic keys. *Proceedings of the National Computer Conference*, 48(1979), 313–317.

[5] Erwin Kreyszig, *Advanced Engineering Mathematics Eighth Edition*, John Wiley and Sons, INC , (2010), 250–258.

⋆⋆⋆⋆⋆⋆⋆⋆⋆
ISSN(P):2319 – 3786
Malaya Journal of Matematik
ISSN(O):2321 – 5666
⋆⋆⋆⋆⋆⋆⋆⋆⋆