Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 10, Issue, 07, pp.71759-71763, July, 2018

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

ISSN: 0975-833X

RESEARCH ARTICLE

MODIFICATION OF AN ENCRYPTION SCHEME BASED ON THE LAPLACE TRANSFORM

***Roberto P. Briones**

School of Mathematical and Computer Sciences, Heriot-Watt University Malaysia

ARTICLE INFO

Article History:
Received 22nd April, 2018
Received in revised form
06th May, 2018
Accepted 20th June, 2018
Published online 31st July, 2018

Key words:
Laplace transform, Maclaurin expansion, Plaintext, Cyphertext, Security key, Modular arithmetic.

ABSTRACT

Hiwarekar (2012) recently introduced a new scheme in cryptography the construction of which is based on the Laplace transform of the Maclaurin series of a C^{∞} function t^k *(rt)*. (Gupta and Mishra, 2014) posit that the single-iteration procedure offers a weak encryption scheme by showing that cyphertext messages can be decrypted by elementary modular arithmetic, and stating that the procedure is independent of the Laplace transform. This paper examines the conditions that give rise to the encryption scheme based on the Laplace transform, and will discuss ways of strengthening the purported sources of weakness of such cryptographic process. A modification of the initial step of the encryption scheme is then offered, giving rise to two passwords for a single iteration, hence increasing the security of the encryption.

Citation: Roberto P. Briones. 2018. "Modification of an encryption scheme based on the laplace transform", *International Journal of Current Research*, 10, (07), 71759-71763.

INTRODUCTION

Hiwarekar (2012) recently introduced a new scheme in cryptography whose construction is based on the Laplace transform. The encryption process is based on pre-selecting an underlying C^{∞} function *f(rt)*, writing out its Maclaurin series, multiplying said series with t^k, multiplying term-wise the numerical codes of the letters of a plaintext message with the coefficients of the first terms of the previous series, and then determining the Laplace transform of the subsequent finite series, with a view of utilizing the resulting coefficients of the last series as the basis of the cyphertext. Gupta and Mishra (2014) have shown that for a pre-set function te^{rt} using single iteration, and in general for the class of functions $tf(t)$), (1) simple cyphertext messages can be decrypted by elementary modular arithmetical arguments, and (2) the encryption method is independent of the Laplace transform definition. In the end the paper concludes that the single-iteration Hiwarekar cryptographic process is a "weak" scheme. This paper will offer a critique of the conditions that give rise to the encryption scheme based on the Laplace transform, and will discuss ways of strengthening the purported sources of weakness of such cryptographic process. A broader definition of the encryption scheme is offered afterwards, based on a two-password system for a single iteration, which can be generalized for multiple iterations, thereby increasing the security of the encryption.

The Coding Scheme: Definition and Example

The idea behind the encryption is expressing the plaintext as a vector $\vec{v} = \langle v_1, v_2, ..., v_n \rangle$, the vector of the function $t^n g(t)$ as $\vec{w} = \langle w_1, w_2, ..., w_n \rangle$ and then getting their "star" product $\vec{v} * \vec{w} = \langle v_1 w_1, v_2 w_2, ..., v_n w_n \rangle$, where n is the number of letters in the plaintext. The components of $\vec{w}$ are determined from the coefficients of the Laplace Transform of the Maclaurin expansion terms of a pre-chosen function $t^k f(rt)$, where $f(t)$ is some infinitely differentiable function whose Maclaurin expansion has positive coefficients.

Corresponding author:* **Roberto P. Briones
School of Mathematical and Computer Sciences, Heriot-Watt University Malaysia
DOI: https://doi.org/10.24941/ijcr.31417.07.2018

The cyphertext for a plaintext is then formed by corresponding the letter with its order in the alphabet. Without loss of generalization, we can define the substitution function ϕ so that $\phi(A)=0,\ \phi(\mathrm{B})=1,\ \phi(\mathrm{C})=2,\ \ldots,\phi(\mathrm{Z})=25$. A pre-selected number *m* is added to each component after the star product. (Hiwarekar, 2015) gave an encryption example by encoding the word "FLOWER". The process is laid out below, under the initial assumption that both sender and receiver have agreed upon a security key, which is produced at the last step. The receiver is also separately informed about the other parameters *k, r, m,* and the function *f(t).*

Step 1. "FLOWER" corresponds to the vector $\vec{v}=\langle 5,11,14,22,4,17\rangle$.

Step 2. $t^2\sinh 2t=\sum_{i=0}^{\infty}\frac{2^{2i+1}t^{2i+3}}{(2i+1)!}=2t^3+\frac{2^3t^5}{3!}+\frac{2^5t^7}{5!}+\frac{2^7t^9}{7!}+\cdots+\frac{2^{2i+1}t^{2i+3}}{(2i+1)!}+\cdots$

Since the plaintext has six letters, the expansion is stopped after the sixth term, and resulting function becomes the new function $g(t)$ for which we will get the Laplace transform. Hence

$g(t)=2t^3+\frac{2^3t^5}{3!}+\frac{2^5t^7}{5!}+\frac{2^7t^9}{7!}+\frac{2^9t^{11}}{9!}+\frac{2^{11}t^{13}}{11!}$, and

$L(g(t))=\frac{12}{s^4}+\frac{160}{s^6}+\frac{1344}{s^8}+\frac{9216}{s^{10}}+\frac{56320}{s^{12}}+\frac{319488}{s^{14}}$, thus

$\vec{w}=G_0=\langle G_{0,0},G_{1,0},G_{2,0},G_{3,0},G_{4,0},G_{5,0}\rangle=\langle 12,160,1344,9216,56320,319488\rangle$

The zero in the subscript of the *G*'s imply that the encryption is at the zeroth iteration of the encryption process.

Step 3.

Thisimplies that $\vec{v}*\vec{w}=\langle 60,1760,18816,202752,225280,5431296\rangle$, whence

$\vec{v}*\vec{w}+\vec{c}=\langle 65,1765,18821,202757,225285,5431301\rangle$, where $\vec{c}$ is the constant vector $\langle 5,5,5,5,5\rangle$

Now take modulo 26 of each of the components:

$65\equiv 13 \bmod 26$ (Quotient is 2)
$1765\equiv 23 \bmod 26$ (Quotient is 67)
$18821\equiv 23 \bmod 26$ (Quotient is 723)
$202757\equiv 9 \bmod 26$ (Quotient is 7798)
$225285\equiv 21 \bmod 26$ (Quotient is 8664)
$5431301\equiv 5 \bmod 26$ (Quotient is 208896)

Step 4. Modulo 26 then, the vector $G_1=\langle G_{0,1},G_{1,1},G_{2,1},G_{3,1},G_{4,1},G_{5,1}\rangle=\langle 13,23,23,9,21,5\rangle$. The corresponding cyphertext is thus "NXXJVF". The sender sends the security key $\vec{k}=\langle 2,67,723,7798,8664,208896\rangle$, whose components are the quotients obtained in the modular arithmetic performed in the previous step.

3 Critique of the Encryption Algorithm

Using the same example given by Hiwarekar, (Gençoğlu, 2017) used the relation

$G_{l,0}\cdot w_l+5=26k_l+G_{l,1}$

to find the components of G_0 given the vector G_1 is known, by reversing the modular arithmetic. If the attacker intercepts the coordinates k_l of the security key, then G_0 can be found from the prior knowledge of w_l and $G_{l,1}$. In the same example above, using the formula

$$G_{l,0} = \frac{26k_l + G_{l,1} - 5}{w_l},$$

values can be easily found as follows:

$$G_{0,0} = \frac{26k_0 + G_{0,1} - 5}{w_0} = \frac{26 \times 2 + 13 - 5}{12} = 5,$$

$G_{1,0} = 11, G_{2,0} = 14$, $G_{3,0} = 22, G_{4,0} = 4$, and $G_{5,0} = 17$.

In the same vein, (Mishra and Gupta, 2014) encodes the plaintext "PROFESSOR" to get the cyphertext "PQMEIYEKM", using the parameters k = 1, r = 2, m = 0, and $f(t) = e^t$, thereby using the Laplace transform of the sum of the first nine terms of te^{2t}, namely

$$\frac{15}{s^2} + \frac{68}{s^3} + \frac{168}{s^4} + \frac{160}{s^5} + \frac{320}{s^6} + \frac{3456}{s^7} + \frac{8064}{s^8} + \frac{14336}{s^9} + \frac{39168}{s^{10}}$$

Using modular arithmetic, they then proceeded to decode letter by letter in the same fashion as (Gençoğlu, 2017). Mishra and Gupta even went deeper as to say that the manner the encoding was done was actually independent of the Laplace transform. Although they questioned whether the parameter *r* was part of the key or a pre-shared value, they carried their decoding using modular arithmetic as if the parameters *k*, *r*, and $f(t)$ were pre-shared, and somehow intercepted by the potential third-party attacker. In deciphering the cyphertext "PQMEIYEKM", they didn't even assume that they got hold of the security key, and deciphered the cyphertext at face value using brute force with modular arithmetic. Once possible candidates were laid out, it becomes a matter of arriving at the recognizable word to finally get the original plaintext.

In addition, letting $P_1P_2P_3P_4P_5$ be the plaintext and $C_1C_2C_3C_4C_5$ the cyphertext, Mishra and Gupta (2014) showed that

$$L\left\{\sum_{i=1}^{n} \frac{r^{i-1}\phi(P_i)t^i}{(i-1)!}\right\} = \sum_{i=1}^{n} \frac{ir^{i-1}\phi(P_i)}{s^{i+1}},$$

which produced the value $G_{l,1}$. (In their encryption, $m = 0$.)The *i*th letter from the cyphertext is then obtained by evaluating the expression $C_i = \phi^{-1}\left\{ir^{i-1}\phi(P_i) \bmod 26\right\}$. Since the bracketed expression was dependent only on *r* and ϕ, they were prompted to conclude that the Hiwarekar's encryption algorithm is independent of the Laplace transform.

4. Counterarguments

For a single iteration the Hiwarekar encryption algorithm is straightforward to decode using modular arithmetic. However, in both examples in the preceding section, it was assumed that the values of the parameters *m, k, r,*and even the function $f(t)$ were known in advance to a third-party attacker prior to decoding, hence pre-shared. This fact would render the breaking of the cyphertext relatively easy if the third-party attacker could somehow intercept the security key. In addition, the computation's complexity in the preceding examples was ameliorated by the fact that only a single iteration was used, although the complexity would drastically increase if multiple iterations are performed. Indeed, Hiwarekar (2014) talked about encrypting the original plaintext $P_0P_1P_2\cdots P_{n-1}$ to the cyphertext $\phi^{-1}(G_{0,j}G_{1,j}G_{2,j}\ldots G_{n-1,j})$, where the resulting cyphertexts are encoded up to jiterations.

In the first example, for instance, two iterations under the same parameters would initially encode "FLOWER" to "NXXJVF" which in turn would encode to "FTDJLF". The computations are as follows:

$$\vec{c}_1 = NXXJVF = \langle 13, 23, 23, 9, 21, 5\rangle$$
$$\vec{w} = \langle 12, 160, 1344, 9216, 56320, 319488\rangle$$
$$\Rightarrow \vec{c}_1 * \vec{w} + \vec{c} = \langle 161, 3685, 30917, 82949, 1182725, 1597445\rangle,$$

and then

$161 \equiv 5 \bmod 26$ (Quotient is 6)

$3685 \equiv 19 \bmod 26$ (Quotient is 141)
$30917 \equiv 3 \bmod 26$ (Quotient is 1189)
$82949 \equiv 9 \bmod 26$ (Quotient is 3190)
$1182725 \equiv 11 \bmod 26$ (Quotient is 45489)
$1597445 \equiv 5 \bmod 26$ (Quotient is 61440)

This implies the cyphertext "FTDJLF", with security key $\langle 6, 141, 1189, 3190, 45489, 61440 \rangle$. A third iteration, j = 3, would give the cyphertext "NDHJXF".In general, for j iterations of the plaintext $P_1 P_2 P_3 \cdots P_n$, to decode from $G_{i,j}$ to $G_{i-1,j}$, one would use the formula

$G_{i,j} = \{G_{i,j-1} \kappa_r(i) + p\} \bmod 26 = q_{i,j} - 26k_{i,j}$,

where $i = 1,2,3,..., n$. $\langle k_{1,j}, k_{2,j}, k_{3,j}, \ldots, k_{n,j} \rangle$ is the security key at the (last) jth iteration. The coefficient function $\kappa_r(i)$ depends only on i (once r is already set) and depends on the Laplace transform of the Maclaurin expansion of the selected function $f(t)$ (Hiwarekar, 2015).

If decoding every letter is performed using modular arithmetic beginning at the last iteration, the problem of determining which letter is actually the one intended by the sender arises, as there might be at least two possible candidates that may arise during the solution of a modular equation. This could result in letters forming intermediate cyphertext words that may look meaningless or not found in the dictionary.

5. Modification of the Encryption Algorithm

In view of the fact that much of the criticism of the Laplace transform-based encryption algorithm is based on single – iteration, this paper proposes a modification of the said encoding scheme, particularly at the initial step. A weakness of the Hiwarekar scheme is that the coefficients of the first n terms of the Maclaurin expansion are used. The modification recognizes the fact that the Maclaurin coefficients of a $C^{\infty}(R)$ function $f(t)$ is infinite in number, and hence the n coefficients can be *randomly* chosen from any of the infinite terms in the series. This gives rise to a second security, which is random by nature and depends on the choice of the sender. This additional key will come from subscripts selected by the sender from the infinite series. Without this subscript key, it becomes much harder to break the cyphertext, even if the security key is intercepted.

Example

Choose $f(t) = e^t$, $k = 1, r = 2, m = 3$, so that the function whose Laplace transform will be considered is $g(t) = te^{2t}$. We want to encode the plaintext "SECRET" under the modified procedure.

Step 1

"SECRET" would be defined by the vector $\vec{v} = \langle 18, 4, 2, 17, 4, 19 \rangle$.

Step 2

$$te^{2t} = \sum_{i=0}^{\infty} \frac{2^i t^{i+1}}{i!} = t + 2t^2 + 2t^3 + \frac{8}{6}t^4 + \frac{16}{24}t^5 + \frac{32}{120}t^6 + \cdots$$

Now randomly choose six values of the subscript *i*and take their corresponding terms. Suppose the *i*values are 1, 5, 11, 17, 20, and 23. We then take the terms that are defined by the selected *i* values, and add them up to form the function

$$\frac{2^1 t^2}{1!} + \frac{2^5 t^6}{5!} + \frac{2^{11} t^{12}}{11!} + \frac{2^{17} t^{18}}{17!} + \frac{2^{20} t^{21}}{20!} + \frac{2^{23} t^{24}}{23!}$$

$$\Rightarrow L\left\{\frac{2^1 t^2}{1!} + \frac{2^5 t^6}{5!} + \frac{2^{11} t^{12}}{11!} + \frac{2^{17} t^{18}}{17!} + \frac{2^{20} t^{21}}{20!} + \frac{2^{23} t^{24}}{23!}\right\}$$

$$= \frac{2^1}{1!} \cdot \frac{2!}{s^3} + \frac{2^5}{5!} \cdot \frac{6!}{s^7} + \frac{2^{11}}{11!} \cdot \frac{12!}{s^{13}} + \frac{2^{17}}{17!} \cdot \frac{18!}{s^{19}} + \frac{2^{20}}{20!} \cdot \frac{21!}{s^{22}} + \frac{2^{23}}{23!} \cdot \frac{24!}{s^{25}}$$

$$= \frac{4}{s^3} + \frac{192}{s^7} + \frac{24576}{s^{13}} + \frac{2359296}{s^{19}} + \frac{22020096}{s^{22}} + \frac{201326592}{s^{27}}$$

$$\Rightarrow \vec{w} = \langle 4,\ 192,\ 24576,\ 2359296,\ 22020096,\ 201326592 \rangle$$

Step 3

This implies that

$$\Rightarrow \vec{v} * \vec{w} + \vec{c} = \langle 75,\ 771,\ 49155,\ 40108035,\ 88080387,\ 3825205251 \rangle$$

Step 4

$75 \equiv 23 \bmod 26$ (Quotient is 2)
$771 \equiv 17 \bmod 26$ (Quotient is 29)
$49155 \equiv 15 \bmod 26$ (Quotient is 1890)
$40108035 \equiv 19 \bmod 26$ (Quotient is 1542616)
$88080387 \equiv 5 \bmod 26$ (Quotient is 3387707)
$3825205251 \equiv 23 \bmod 26$ (Quotient is 147123278)

The corresponding cyphertext is thus "XRPTFX". The sender then sends the security key $\vec{k} = \langle 2,\ 29,\ 1890,\ 1542616,\ 3387707,\ 147123278 \rangle$ and the subscript key $\vec{s} = \langle 1,\ 5,\ 11,\ 17,\ 20, 23 \rangle$ to the receiver to be able to determine the placement of the terms in the series (which specifies the powers of *t*).

6. Conclusion

The addition of a subscript key $\vec{s}$ enhances the security of the single-iteration Hiwarekar encryption algorithm. The random nature of the components of the subscript key affords the cyphertext of an additional layer of security that is difficult to break as long as the parameters of the encryption are pre-arranged to be known only to the sender and the receiver.

REFERENCES

Gençoğlu, M.T. 2017. Cryptanalysis of a New Method of Cryptography using Laplace Transform Hyperbolic Functions. Communications in Mathematics and Applications8 (2), 183–189.

Gupta, P., Mishra, P.R. 2014. Cryptanalysis of "A New Method of Cryptography Using Laplace Transform". In: Pant, M., Deep, K., Nagar, A., Bansal, J., (eds) Proceedings of the Third International Conference on Soft Computing for Problem Solving. Advances in Intelligent Systems and Computing 258, 539 – 546. Springer, New Delhi.

Hiwarekar, A.P. 2012. A new method of cryptography using Laplace Transform. *International Journal of Mathematical Archive* 3 (3), 1193 – 1197.

Hiwarekar, A.P. 2015. Application of Laplace Transform for Cryptography. *International Journal of Engineering & Science Research* 5 (4), 129 – 135.

****