LIE-Let it Encrypt: An Encryption Algorithm meant for Secure Transactions

Mukta Sharma, Dr. R.B. Garg

Abstract— **Today in the era of technology; security plays an important role especially when it comes to online transactions. There are various threats via which the victim is attacked and the data is lost or misused [8][5][[4][9]. Many counter measures have been taken like biometric, SSL, HTTPS, Intrusion detection system (IDS), Cryptography etc. According to [8] CIA Triad, Confidentiality, availability, and integrity are the three main objectives to provide security.**

Biometric ensures that only authentic user can access the data, by scanning the finger, iris, etc. Similarly, the idea behind the popular protocol SSL is to safeguard the path for traversing the data. Https ensures the safety of every transaction. Whereas, cryptography ensures that the data should be secure even on the insecure path, hacked or accessed by an even unauthentic user. Cryptography deals with securing the data and fulfilling all the three above mentioned objectives.

The paper is written with an objective to not only discuss cryptography but also discuss LIE- an encryption algorithm, how the encryption algorithm is tested, etc. LIE is tested on various parameters like time, space, desirable properties etc.

Index Terms— **CIA; LIE; TRNGs; OSI &Symmetric Key Encryption Algorithm**

I. INTRODUCTION

Information technology has made a dream come true, it has actually made it possible to not only hear but also see a person sitting millions of miles away. The facilities offered by the technology have transformed the way of thinking, interacting, conducting business, studying, etc. In almost all domains it has changed and helped the society to grow tremendously. It is a well-known fact that every coin has two sides; similarly, the technology has its own set of prejudices. Besides the health issues caused due to extensive use of laptop, mobiles, i-pods, i-pads etc.; the fear of theft of data is another important matter of concern. The data which is kept at stake could be images, videos, chats, mails or may be bank details etc. which might be compromised. Many researchers are extensively working on enhancing the security in areas like anti-virus, anti-phishing, biometric, cryptography etc.

A. *Classification of security goals*- According to CIA Triad [8], following three objectives is security goals.

- Confidentiality- information should be confidential not in terms of storage of information but also while transmission of information. The information should be shown to only authentic users and if a hacker hacks the data; some unreadable format should be shown so that the original message is safe.
- Integrity-Information should be original, complete, uncorrupted. In short, information should not tamper.
- Availability-Information should be available whenever an authentic user needs it.

Cryptography- Kryptos (Secret) and Graphite (writing) two Greek words, meaning Secret Writing. Cryptography is meant to save the data while transacting online. Protocols like SSL, HTTPS, PGP, TLS, etc. use the encryption algorithm for safeguarding the transaction. Cryptography is a technique used to convert messages to jumbled text or unreadable messages [1]. Cryptography is a subset of cryptology (Study which deals with reading, writing and breaching of the code. Cryptography can be subdivided into Asymmetric and Symmetric Key.

- The asymmetric or public key is based on the concept of using a pair of a key; one public key is globally known to all and one private key used to decipher the code. Whereas, the Symmetric key is a concept based on the single key; commonly used by both the parties (receiver/sender). Symmetric key solves the first objective of confidentiality, privacy. Asymmetric key focuses on integrity, availability, non-repudiation of data.
- Symmetric key algorithms can be designed based on how the data block is input and processed. If the data is accepted in a fixed block size and processing is done on that block then it can be said as Block Ciphers. Examples of Block ciphers are DES, 3DES, AES, IDEA, Blowfish etc. If one bit or byte is processed at a time it is said as Stream Ciphers or state cipher, RC4 is a popularly known algorithm of a stream cipher, a one-time pad is another example of the Stream cipher.

II. LIE FLOW

Let it Encrypt, is a block cipher which accepts [6] 128 bits block as an input. It uses 256 bits key for ensuring security. It uses Feistel network and performs round function; which iterates eight times. It uses 8 sub-keys of 64 bit; which are generated from the key itself, therefore; 4 sub-keys are discretely unique and rest 4 sub-keys have some redundant values. The sub-keys have been designed carefully keeping security as the key factor. LIE is based on Claude Shannon principle of confusion-diffusion and therefore uses three permutation matrixes for better security.

A.Key Generation-

It is based on a random number generated by rolling dice [7]. This number will iterate 256 times and generates a number for each location for the key. This will generate number only from 1 to 6. Thus, the output of this function will be checked for being an even number or odd number. If a number is even then “0” will be placed at the location “I” of key matrix else

Mukta Sharma, College of Computing Science and Information Technology, TMU, Moradabad, India
Dr. R.B. Garg, Department of Computer Science, University of Delhi, Delhi, India

"1" will be entered. This gives equal probability to both "0" and "1" to be assigned to any location in the key array and also offers a well-known randomness of rolling dice. Later the key was used to generate sub-keys. The key size is 256 bits, eight sub-keys sizes of 64 bits are generated. Four completely discrete sub-keys were designed and four sub-keys with some redundant values were designed for each round.

B. Pseudo code [7]

Step 1: Initialize key matrix: key [] = 0
Step 2: For I 0 -> 256
Step 3: num = Get random number between 1 to 6.
Step 4: if num%2 ==0
Step 5: key[i] =0
Step 6: Else key[i] =1
Step 7: i->i+1
Step 8: REPEAT UNTIL i<256
Step 9: Generate sub keys k1 to k8 using sub keys matrices
Step 10: Take 128 bit plain text as input ->PT
Step 11: Perform Initial Permutation using Table 1
Step 12: While i <>8
Step 13: Divide PT (128 bits) into L0 & R0 each 64 bit.
Step 14: Li = Ri-1
Step 15: Ri = F (Li-1, ki)
F (Li-1, ki)
Permute Li-1 using table 2 Inner Permutation
Perform Left Circular Shift
Li-1 XOR ki
Step 16: I-> i+1
Step 17: Obtain CT' = R8L8
Step 18: CT = Perform Final Permutation using Table 3
Step 19: CT=CT*k

C. Flowchart [6]

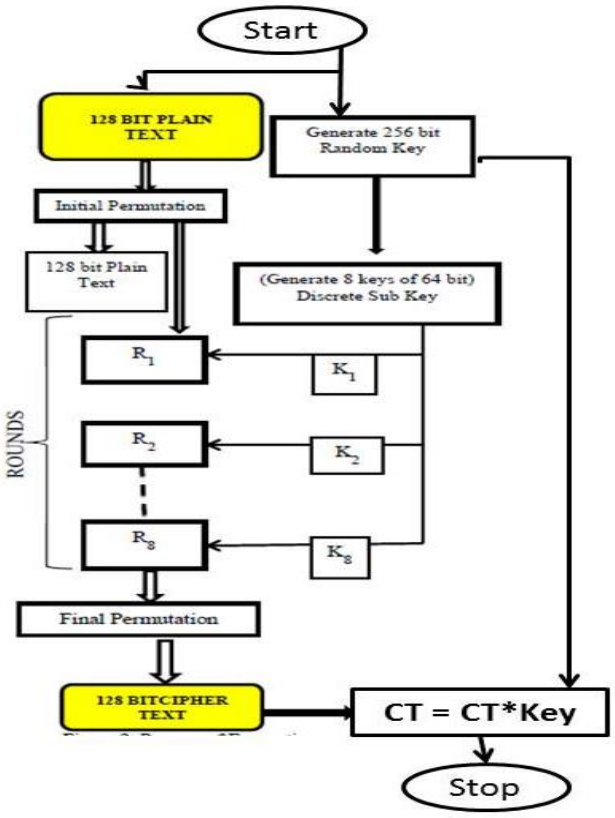

FIG.1.A STRUCTURE OF LIE

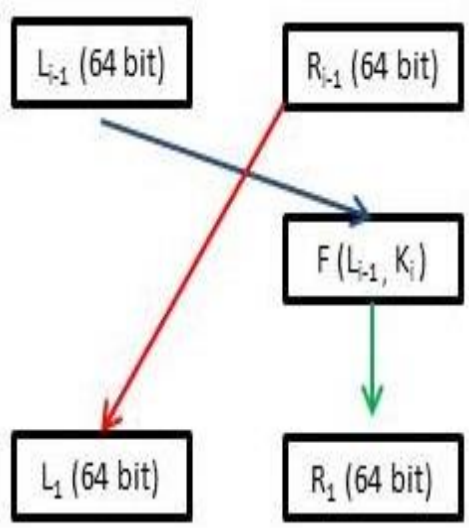

FIG.1.B FEISTEL NETWORK

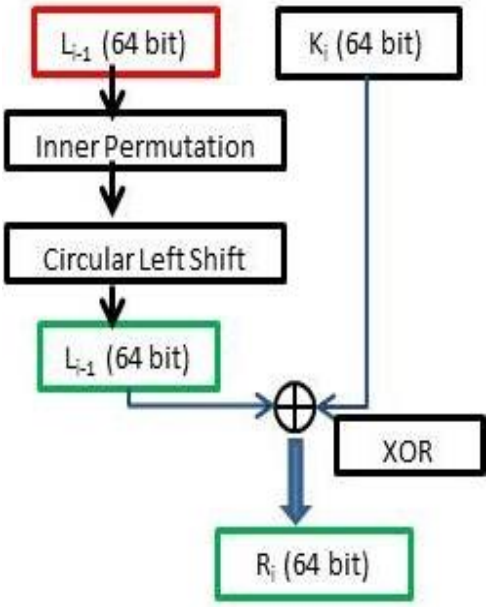

FIG.1.C ROUND FUNCTION (F)

III. TESTING LIE ON VARIOUS PARAMETERS

LIE is a symmetric key block algorithm, and as all cryptography algorithms are measured on the following parameters:

A. **Architecture**- As the name suggests it deals with the design or basic organization, functionality of an algorithm.

LIE was designed by *Mukta Sharma* and *R.B. Garg* in the year *2016*. The key size is an essential element of any encryption algorithm. LIE uses *256* bits keys, which is based on the concept of *True Random Number Generators*. It uses the concept of Rolling Dice if the number generated randomly is an even number it will be replaced with "zero (0)" else with "one (1)". 8 sub-keys are generated for 8 rounds (or

iterations). It has been stated by Shannon to ensure security; four rounds are sufficient if the keys are unique. This algorithm iterates 8 times and with 4 completely discrete and 4 partially unique sub-keys. The Algorithm is designed on Feistel Network and the block size is 128 bits. The algorithm comprises of three Permutation Matrixes (Initial Permutation, Inner Permutation, and Inverse Permutation matrix) to implement Diffusion (Claude Shannon concept for better security). The algorithm performs operations like XOR, <<<, *.

B. **Scalability** – is an essential parameter as it shows the proficiency and fitness of a system, network or process when it grows. In LIE scalability is checked with the computational power, throughput time, Memory consumption, Encryption/Decryption rate etc.

i. Encryption Time/ Rate- The time is taken for Encryption Process (converting plain text to cipher text) depends on the processor's speed, the complexity of an algorithm, and hardware used such as main memory etc. LIE Encryption rate is high; it takes less time for enciphering the data.

ii. Decryption Time- The time is taken for Decryption (converting cipher text to plain text). LIE Decryption rate is high; it takes less time for deciphering the data.

iii. Throughput Time- Total plain text bytes/Total encryption time. LIE throughput time is also very less and throughput is very high.

iv. Memory Usage- Depending on the number of functions the total memory is consumed during the entire process. LIE consumes less memory and less power consumption.

v. CPU Process Time- Time while a CPU is dedicated for a particular procedure. The CPU process time is very less and Computational speed is very fast.

C. **Security**- one of the prime factors to test cryptography algorithm is security. It depends on varied factors which make an algorithm safe like confusion/diffusion/ key length etc.

i. Differential Cryptanalysis- It can be attained by 2^Block Size. In the case of Lie, it is 2^128.

ii. Processing Complexity- It can be calculated by 2^KeySize. LIE processing complexity is 2^256.

iii. Cryptanalysis Resistance- LIE is vulnerable to weak keys.

D. **Flexibility**- As the name says, flexibility proposes with the ability to change or modify. Here, the idea is to determine whether the algorithm is able to endure minor modifications in key size, block size, the number of rounds etc. or are they static. LIE algorithm is not flexible in terms of Key, i.e. 256 bits and rounds i.e. 8. It accepts 128-bit block size, in case a user input less than 128 bit it will automatically expand it to 128 bits. The Ciphertext or the output block is variable.

E. **Limitations (Known Attacks) –** Will shed light on the vulnerable areas; where an attacker can breach or where the data can be compromised. LIE is vulnerable to weak keys.

IV. LIE Based on Time

Number of letters (Block Size in bytes)	Encryption time (ns)	Decryption Time (ns)
16	21254634	41067686

Table 1: Time Taken

V. LIE Based on Space

	Bytes	KB	MB
LIE Encryption Space	8295480	8101.0546875	7.911186218261719MB
LIE Decryption Space	7112208	6945.515625	6.7827301025390625

Table 2: Space Used

VI. LIE Based on desired properties

A. **Avalanche-** It is tested by altering or flipping a single bit either in the plaintext or the key. As demonstrated below, LIE fulfils more than 50% flip of bits.

i. The strict avalanche criterion (SAC) guarantees a one-bit change in key or plaintext will lead 50% of the output to be flipped [3].

ii. The bit independence criterion (BIC) depicts that inversion of 'i' a single input bit will reflect the change in i, j and k output bits independently.

Plain Text: Son

Output Cipher:

524010112456565268043433051154071317042941511903510010772706098491125579126575859654472938107266801494769289654645 5

****

Plain Text: Sun

Output Cipher:

524010112456565266325352683361233921351397515982933270693985683738809323097316153203150824514374773074429694930583

Instead of Son, the message is altered as Sun and there is 82.6% flip in bits by changing just one byte. (95/115 = 82.6%)

B. **Completeness**- According to encryption, completeness is a necessary property. Completeness depicts that every output/ cipher bit is dependent on every input/plain text bit [2]. Change in a single bit of the plain text/input might lead to a change in every bit of the output/cipher text. The average chance of changing is 50%.

Let us imagine change made in one byte of the plaintext would affect only 1 byte of the Ciphertext. It would be very easy for an attacker to guess different plaintext-Ciphertext pairs. These days it is very convenient to guess the plain text-cipher text as the standard protocols use regular commands (e.g. "get," "put," "mail from," etc.). In such case, the intruder would need to collect 2^64 (~1020) plaintext-Ciphertext pairs to break the cipher.

C. **Statistical Independence**- It indicates that the input and output should not be statistically dependent on each other. The beauty of the algorithm lies in this property. As this property emphasis on keeping no statistical relation between the input and output to ensure the safety of plain text, even when the hacker has compromised the whole of the cipher. The properties like Confusion & Diffusion used to build encryption algorithms give a critical point around it.

VII. Conclusion

Symmetric key cryptography has been used for centuries even before the use of computers. Encryption algorithms are preferred for secure data transaction. LIE was designed to ensure and provide better security with consuming less time and space. The paper depicts the algorithm flow, along with the pseudo code. The paper depicts the butterfly effect; by flipping one byte actually affect more than 50% of the output (Ciphertext). Paper has shed light on the parameters used to evaluate any encryption algorithm.

References

[1] Govinda, K. & Sathiyamoorth, .E. (2011). Multilevel Cryptography Technique Using Graceful Codes. Journal of Global Research in Computer Science. Volume 2(7)

[2] Ramanujam, S., & Karuppiah, M. (2011). Designing an algorithm with high Avalanche Effect. International Journal of Computer Science and Network Security. 11(1).

[3] Schneier, B. (1994).Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish),‖ Fast Software Encryption, Cambridge Security Workshop Proceedings, Springer-Verlag, 1994, Available at http://www.schneier.com/paper-blowfish-fse.html

[4] Sharma, M., & Dwivedi, S. (2013). Hurdles for the Online Security Breach. International Conference organized by MERI. Later published in in refereed journal of MERI, Vol. 6(2). pp. 85-92.

[5] Sharma, M., & Garg, R.B. (2012). Security Apprehension to E-banking sector. International Conference organized by TMU.

[6] Sharma, M., Dwivedi, S., and Garg. R.B. (2015). "Let It Encrypt (LIE)". International Journal of Computer Applications, 128(8), pp. 9-15

[7] Sharma, M., Garg, R.B. (2016). "Comparative Study of Enhanced LIE, NPN, & DES Algorithm". International Journal of Computer Science and Information Security (IJCSIS), Vol. 14(2).

[8] Stallings, W. (2011). Cryptography and Network Security: Principles and Practice, 5th Edition. US, USA: Pearson Education, Prentice Hall.

[9] Whitman, M. E., and Mattord, H. J. (2015). Principles of Information Security, 5th Edition. Cengage Learning. Boston. Available at: http://works.bepress.com/herbert_mattord/32/

Mukta Sharma, Research Scholar- She holds M. Phil (Computer Science), M. Tech (IT), M.Sc. (CS), PGDCA. Currently she is pursuing Ph.D. in Computer Science from Teerthanker Mahaveer University, Moradabad. She has more than fourteen years' experience of teaching in various undergraduate & postgraduate courses of Indian Universities like DTU, GGSIP University, MCRP, Kurukshetra, IASE, PTU & UPTU etc. She has co-authored a book titled "Web Technologies: Planning, Designing and Development of Websites", Galgotia Publications Ltd. She has contributed a chapter "Services of Mobile Commerce", in Securing Transactions and Payment Systems for M-Commerce, IGI Global, and ISSN: 1935-2700 with Advances in E-Business Research (AEBR) book series, Lee (Western Illinois University, USA). To her credit she has various research papers published in national and international Journals/ conferences like IEEE, IJCSIS, Indian Publisher, IJCA, IJATES etc. She has contributed various book reviews.

Dr. R.B. Garg, Research Guide obtained his Graduation in Mathematics, Masters in Statistics and Ph.D. from Delhi University. His research interests include Reliability Theory and Mathematics. He has published more than 45 research papers in International, National journals and Conferences. He has contributed a book title "Contributions to Hardware and Software Reliability" published by World Scientific. He has more than 48 years of experience in academia, industry and administration