Data Partition Based Encryption for Cloud Data Storage

B.Muthulakshmi, M.Venkatesulu

Abstract: Data security gains importance in cloud computing technology which is one of the most beneficial ubiquitous computing. It delivers resources and services via internet by Cloud Service Providers at cloud user's choice of place. Data needs to be outsourced to avail this technology. Outsourced data to cloud is subjected to risks. Hence, researchers developed a lot of security techniques such as encryption schemes, authentication techniques to protect data in cloud. In this paper, a novel Invertible Non-linear Function based Cryptographic System (INFCS) is proposed to make the cloud storage secure and protected. The INFCS model comprises of partitioning, encryption and decryption. In partitioning technique the data of data holder is partitioned or split into number of fragments which are then encrypted using invertible non-linear function. Then they are stored in one or more cloud storage(s). The decryption is performed at the end user's side by doing inverse of invertible non-liner function. Use of the proposed INFCS makes the data more secure and preserved from any unauthorized users and malicious activities. The proposed INFCS is efficient and faster than other existing cryptographic systems.

Keywords: Cloud computing, Data security, Data partitioning, Pseudo plaintext, Invertible nonlinear function and Prime numbers.

I. INTRODUCTION

In recent years the growth of networking technology requires vast computational resources which lead several organizations to outsource their storage (Seny Kamara & Kristin Lauter, 2010). Cloud computing provides the accessibility of computing resources through the Internet. So network users can utilize the available resources on cloud at anywhere and anytime without having the control of anyone (R. Arokia Paul Rajan & S. Shanmugapriyaa (2012) and Murugaboopathi et al (2013)). The cloud computing comprises of several services like Infrastructure as a Service (IaaS), Platform as a s Service (PaaS) and Software as a Service (SaaS). In IaaS the users utilize computing, storage and network infrastructure of service providers whereas in PaaS the users leverages the resources of service providers to execute customized applications. In SaaS the users utilize software which works under the infrastructure of the service providers (G. Clarke (2009) and Hassan Takabi et al (2010)). Hassan Takabi et al (2010) and Richard Chow et al (2009) expressed that assuring privacy and security of cloud environment is critical and having legal issues. There are several security issues present in cloud computing such as backups, data and host security, file system and network traffic (Neha Jain & Gurpreet Kaur (2012)).

This paper proposed INFCS has three phases namely, partitioning, encryption and decryption. In partitioning phase, the data holder partitions his/her data into number of fragments which are then encrypted using Invertible Non-linear Function. This encrypted data is stored in cloud storage with the help of cloud service provider (CSP). When an end user needs this outsourced data, he/she sent a data access request to a data holder. The data holder approves this request by sending key and certificate to end user. With this certificate the end user communicates the CSP to access the cloud data. After the certificate verification process, the CSP command the storage to allow the end user to access the encrypted data. Finally the encrypted data is decrypted with the inverse of non-linear function and obtained key. It provides efficient and fast data operation with reduced cost. The rest of this paper is organized as follows: Section 2 discussed the related works. Section 3 describes problem statement. In section 4 detailed operations of the proposed INFCS are given. In section 5 experimental setup and testing is presented. Section 6 concludes the paper with future work.

II. RELATED WORK

In this section, a literature survey is done for cryptographic techniques. In recent years, different types of cryptographic techniques are proposed by several researchers to make the cloud environment safe and secure for storing confidential or personal data of cloud users. Somdip Dey et al [9] have presented a novel cryptographic technique named SD-AREE which is a modified version of Caesar cipher substitution technique. It also uses a bit manipulation technique along with it for removing redundant data in the plaintext before applying encryption on it. This makes the original text unpredictable for network intruders. In this technique the users align their keys in string format. There are two members such as "code" and "power-ex" are generated by using this key string. The code value is generated by multiplying the ASCII value of every key character with its string length and 2i, where 'i' refers the position of character in the key and it initiates from 0. Add the resultant value of every character obtained from the manipulation. The "power –ex" value is generated by adding all digits of resulted sum. Finally the modulus operation is applied on pseudo code by 16 to obtain the code value. Amit and Bhavesh [10] have presented a new randomized technique for cryptographic operations. The implementation of their randomized and public key generation approach is done by using a protocol and Caesar cipher substitution technique. The implemented algorithm is tested for different sizes of plaintext.

Revised Manuscript Received on June 08, 2019.

B.Muthulakshmi, Department of Computer Applications, Kalasalingam University, Krishnan oil, Tamil Nadu, India.

M.Venkatesulu, Department of Computer Applications, Kalasalingam University, Krishnan oil, Tamil Nadu, India.

The public and private keys are generated by using the correct message at delivery time. Quist-Aphetsi kester [11] presented a hybrid technique for cryptographic operations applying both transposition and classical substitution techniques. Two encryptions are performed here. At first, the plain text is encrypted with columnar transposition cipher and then the resulting cipher text is encrypted with vigenre cipher which produces final encrypted data. To perform columnar transposition, initially a fixed column length key is selected and the plaintext is arranged row by row in the fixed column length table which is equivalent to the string length. Based on the keys, the columns are rearranged in alphabetical order and the message is read column by column. The vigenre substitution takes this resultant cipher text as its key. The process of decryption is performed in reverse form. By applying the strength of columnar transposition, the authors use the strength of vigenre cipher to minimize the drawbacks of each other.

Fadhil et al [12] have combined the properties of public key RSA cryptosystem and knapsack algorithm to propose a hybrid technique for providing highly secure and less complex systems. The proposed technique also necessitates highly secure and less complex systems. When compared to RSA cryptosystem and knapsack algorithm separately this hybrid technique requires very less time to perform the encryption/ decryption operations. The proposed work is divided into two phases. In first phase, the initial encryption of plaintext is done by RSA algorithm and its resultant output given as the input of the knapsack algorithm. The decryption is performed at the receiver side. Cryptography is included to change the transmitting message into other unreadable format for network intruders. Stenography is used to hide secrete of transmitting message into some other messages for making it invisible from the intruders.

Singh et al [13] introduced a novel hybrid scheme by combining the transposition and substitution techniques. At first, individual disadvantages of these techniques are eliminated to develop an uncrackable technique. In this paper, the Caesar cipher substitution and Rail fence transposition are combined to produce a novel and efficient technique. Jiehong et al., [14] have evaluated the performance of AES, Blowfish, and GOST encryption algorithms. Comparison of these three algorithms has been performed at various sizes of data blocks and the comparison results are demonstrated to obtain performances of each algorithm. Among this, the blowfish algorithm provides the better performance regardless of the sizes of plain-text. It is observed that the key expansion process is the weakest segment of this Blowfish algorithm which could consume more time for key expansion than accomplish encryption/ decryption if the size of the plaintext is small. Additionally, they specify that the key expansion of GOST algorithm consumes relatively same amount of time with respect to Blowfish, and it also costs relatively same time for both encryption and decryption to GOST algorithm. Finally, it is concluded that the AES algorithm has longest operation to perform decryption. Wang et al. [15] have realized the arithmetic functions throughout the cipher texts of several users to perform addition and multiplication without the function learning of inputs/ intermediate outputs. However, this technique requires for solving the discrete logarithmic issues that actively limits the input data length. Hence, it isn't appropriate to be implemented into the cases where there are numerous information providers and the given data size is huge. Bhandari et al., [16] have specified that today the most difficult problems in cloud environment are for assuring the data privacy and security of the cloud users. The authors proposed a novel hybrid encryption RSA (HE-RSA) with Advanced Encryption Standard (AES) for efficiency, reliability, and consistency in cloud servers. They expected to utilize different cryptography ideas amid communication with its application in distributed computing and to upgrade the cipher-text security/encrypted data in cloud servers with reducing the utilization of time, cost and memory measure amid for both encryption and decryption. They noticed that the contrast between the execution time of the first RSA and Enhanced Algorithm utilizing Hybrid Encryption-RSA and AES is expanding definitely as the size of exponent is expanding. Ramachandra et al., [17] have surveyed the security challenges in various delivery and deployment models. They have mentioned some attacks of network, hypervisor and hardware and suggested several solutions of checking of insider's attack, secured interfaces and so on.

Here a novel INFCS is proposed for protecting the confidential and sensitive data stored in the cloud from unauthorized end users or any malicious activities. This INFCS provides fast and efficient operations with reduced cost.

III. Problem analysis

The drawbacks of traditional cryptographic mechanism are: 1) it takes high cost and 2) takes more time to perform encryption and decryption operations for data storage in cloud. These limitations are overcome by partitioning the data which offers reduced cost, limited space for storage and better efficiency. Some of the following are privacy and security related issues present in cloud environment.

- ***Access control***: At some level the CSP fails to provide security mechanism which leads the data access of unauthorized users. So intruders can easily access the stored data.
- ***Authentication and Identification***: This problem arises while multiple users requesting data access simultaneously.
- ***Availability***: The requesting cloud service is not available at anytime then this issue will occur.
- ***Unauthorized usage & control policy***: They are considered as the most significant issues in cloud since different CSP have different security mechanisms. At some situations there is a possibility that the CSP has self-interest in client's data or leak them without his/her permission.

IV. PROPOSED SYSTEM

In this paper, a novel Invertible Non-linear Function based Cryptographic System (INFCS) is proposed which makes the encryption more complicated so that the intruders cannot access the encrypted data easily. Decryption is performed by the inverse of invertible non-linear function. Then the integrity of obtained decrypted text is verified.

4.1. System Model

There are for entities such as data holder, cloud service provider, cloud storage and end users involved in the proposed INFCS which are explained below in a detailed manner.

i. Data Holder

Data holder is defined as an entity that is responsible for creating information/data or services need to shared with other end users through the cloud environment. The data holder should secure his/her data from unauthorized users by verifying the certificates all end users. After the verification process a certified authorized user can access the data from the cloud.

ii. Cloud Service Provider

A Cloud Service Provider (CSP) controls the database servers and servers of distributed cloud storage and enable virtual framework to host applications.

- Users can benefit by the resources of providers and make use of it by storing, retrieving and sharing his/her data.
- With the help of user’s authorization the users are allowed to control the data stored in the cloud and this authorization is provided by the client. The functions of this authorization include reading, writing and modifying the data.

iii. Cloud Storage

Cloud storage is defined as a service where user’s data is stored, managed, maintained and backed up remotely. Some of the public cloud storage services like Amazon's S3 and Microsoft's Azure enable their users to move data to the cloud without any cost of constructing and managing a private storage infrastructure. Rather the users make payment to the service providers based on their needs which provide a few advantages such as accessibility and reliability at a minimal cost.

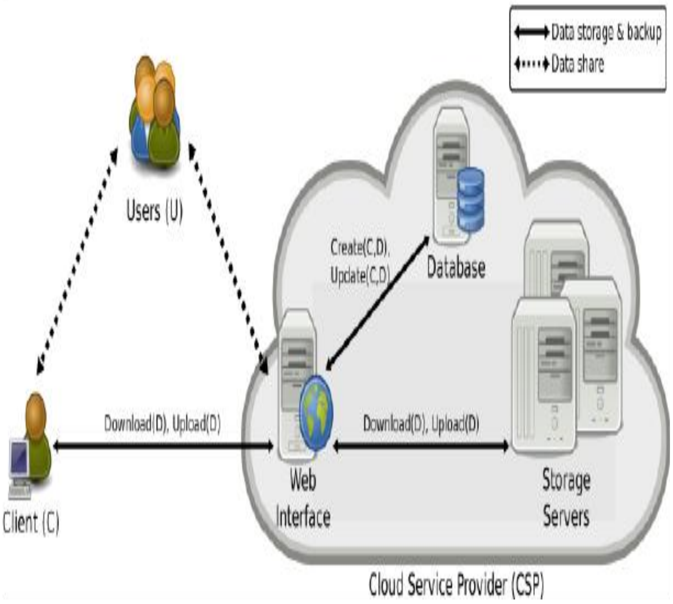


Figure 1: Cloud Architecture

iv. End Users

End users are important entities of cloud that need the data holder services. These services are provided by the CSP after authentication or verification process. If the verification process is successful then the corresponding user can collect encrypted data from the cloud storage and since he/she knows the decryption key, the original data can be decrypted from encrypted data.

Notation	Description
Y_i	Plaintext
i	Number of split or partitioned data $i = 1,2,\Lambda\ ,n$
Y_n	Number of partitioned plaintext
M_i	Pseudo plaintext
G_P	Number of generated primes
P_i, Q_i	Set of prime numbers
P_i^c, Q_i^c	Prime complements
p	Bit of the prime. Here $p = 32$
D_i	Random Integer
d_m	Number of split of random integer
f_i	Generated ciphers
S_i	Sum of encrypted form

Table 1: Notation and descriptions

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

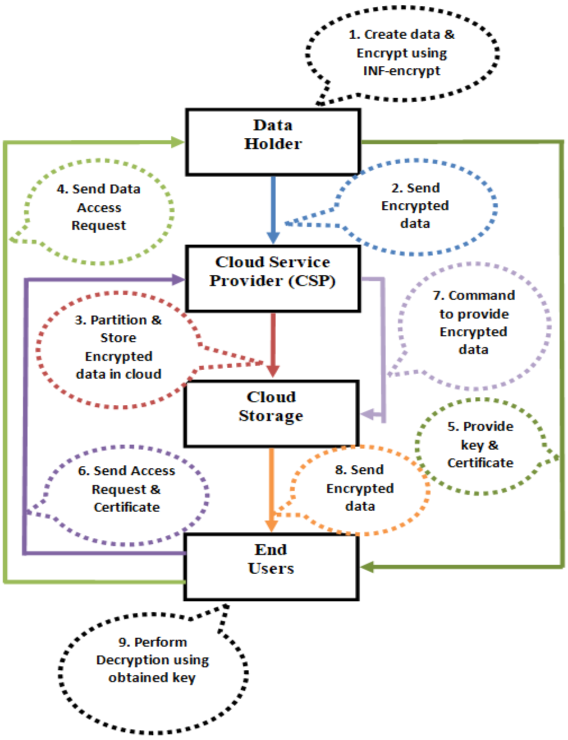

1. Create data & Encrypt using INF-encrypt
Data Holder
4. Send Data Access Request
2. Send Encrypted data
Cloud Service Provider (CSP)
3. Partition & Store Encrypted data in cloud
7. Command to provide Encrypted data
Cloud Storage
5. Provide key & Certificate
6. Send Access Request & Certificate
8. Send Encrypted data
End Users
9. Perform Decryption using obtained key

Figure 2: Proposed flow

The proposed flow is comprised of four entities namely data holder, cloud service provider, cloud storage and end users. At first the data holder partitions and encrypts data using INF-encrypt algorithm. Then multiple encrypted data are stored in either different locations of cloud storage or different cloud storages. The end user sends data access request to the data holder and after processing the request it provides key and authentication certificate. In our proposed INFCS the key is the random integer D_i . But the CSP has the complete control on cloud data only. So the data access request and obtained certificate are also sent to CSP. After certificate verification process CSP commands the storage to provide encrypted data to that particular end user. Finally the end user obtains the encrypted data from the cloud storage and decrypts it using the obtained key from the data holder.

4.2. Partitioning

Data partitioning plays a vital role in INFCS. The original large data is more complex and it is difficult to store it in cloud storage. To reduce storage difficulty and easy data access the large data is split into number of smaller fragments. The partitioned smaller fragments can be stored easily in cloud with reduced time. When there is a necessity the stored data can be quickly accessed from the cloud. The fragmented data is encrypted before storing it in cloud.

4.3. Encryption

Encryption algorithm first partitions the plaintext Y_i into many parts. Each part is converted into pseudo plain text parts by means of an Invertible Non-linear Function as follows,

$$g(x) = ax + b$$

where *a* and *b* are integers and *x* denotes partitioned plain text parts. Each plain text part is multiplied with *a* and then added with *b*. These pseudo plain text parts are referred as M_i . These pseudo plain text parts are subjected to encryption using a set of primes.

Algorithm

Input : Y_i

Method:

i. Convert Y_i into M_i using INF

ii. Generate G_P and $G_P = 2Y_n$

iii. Take P_i, Q_i

iv. Compute P_i^c, Q_i^c [Θ $X_i^c = 2^{p+1} - X_i$]

v. Generate D_i and split it into small integers $D_i = d_1, d_2, \Lambda\ , d_m$ (Here, we take $m = 4$)

vi. $d_n = Y_n$

vii. Create multiple cipher text using the triplets

viii. (P_i, Q_i), (P_i^c, Q_i^c) and $d_1, d_2, \Lambda\ , d_4$ as follows

$$f_{i1} = M_i * P_i * Q_i + d_1$$

$$f_{i2} = M_i * P_i * Q_i^c + d_2$$

$$f_{i3} = M_i * P_i^c * Q_i + d_3$$

$$f_{i4} = M_i * P_i^c * Q_i^c + d_4$$

ix. Store each cipher in different locations of a cloud storage or different cloud storages

4.5. Decryption

In tradition cryptographic system the encrypted data is comprised with decryption keys. But in our proposed INFCS the end user directly communicate the data holder and obtains the decryption key and certificate. The CSP verified this certificate and provide encrypted data. Then the authorized end user can access the encrypted data with knowledge on inverse of invertible non-linear function which given as follows,

$$g^{-1}(x) = \frac{x_i - b}{a}$$

where a and b are integers and *x* denotes partitioned plain text parts.

Algorithm

Step 1: Add all encrypted ciphers

$$S_i = \sum f_{i1} + f_{i2} + f_{i3} + f_{i4}$$

Step 2: Perform $S_i - D_i$ which results M_i with padded zeros
Step 3: Eliminate the padded zeros
Step 4: Apply inverse form of INF
Step 5: Convert binary into byte array
Step 6: Merge all the obtained byte arrays to get the original plaintext

4.6. Advantages of INFCS

In INFCS, the data holder does not provide complete control to CSP on his/her data. If the CSP fails to provide protection mechanism it won't affect the security of data. Because only the encrypted data is stored in cloud and without key and knowledge on inverse INF no one can decrypt the data. Thus it provides complete access control and unauthorized users cannot use the data. Here the CSP executes a simple certificate verification process which requires very less time. If multiple end users requesting data access simultaneously the CSP can manage it. So in this, the authentication and identification problem never arise. The most important issue in cloud environment is control policy which talks about self-interest of CSP on user's data. In this INFCS, the control policy is tightened by not giving complete control to CSP. So the CSP cannot execute unauthorized usage or leak user's data intentionally.

V. Experimental Setup and Testing

The experiment is conducted using Intel(R) Core(TM)2 Duo CPU processor with 4 GB RAM and on Windows 7 platform. The experiment was implemented using Java programming. Figures 4 to 10 show the input, integer conversion of the input, pseudo plain text, set of ciphers, decrypted pseudo plain texts, recovered original text parts in integer format and decrypted original input.

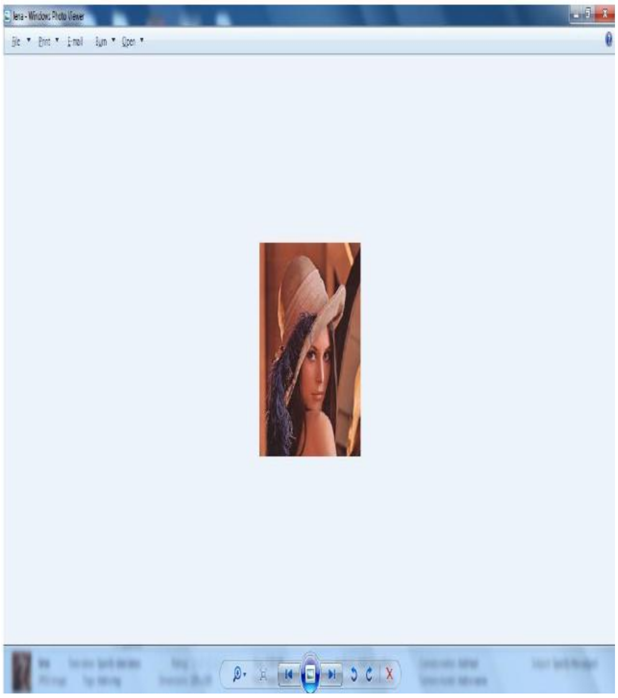
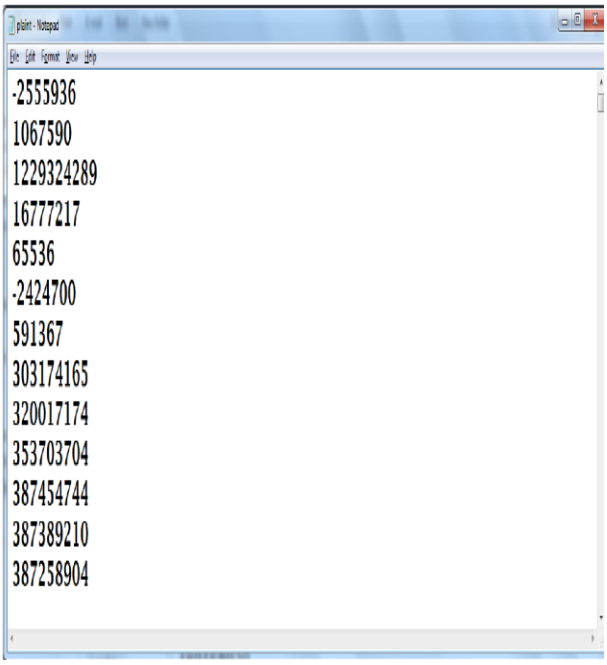
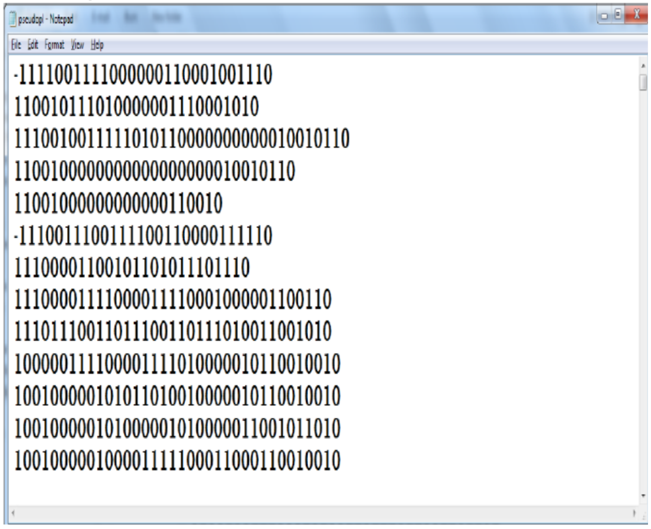


Figure 3: Input file



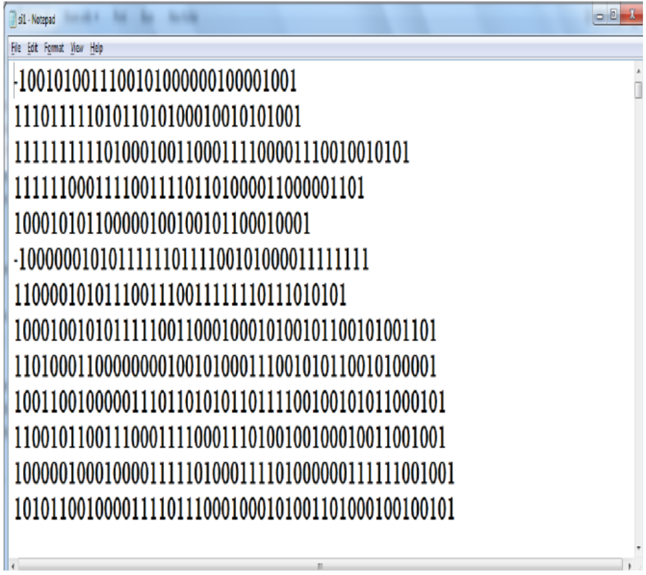
```
-2555936
1067590
1229324289
16777217
65536
-2424700
591367
303174165
320017174
353703704
387454744
387389210
387258904
```

Figure 4: Integer representation of the input file.



```
-1111001111000000110001001110
110010111010000001110001010
1110010011111010110000000000010010110
11001000000000000000000010010110
1100100000000000110010
-1110011100111100110000111110
11100001100101101011101110
111000011110000111100010000001100110
11101110011011100110111010011001010
100000111100001111010000010110010010
100100000101011010010000010110010010
100100000101000001010000011001011010
100100000100001111100011000110010010
```

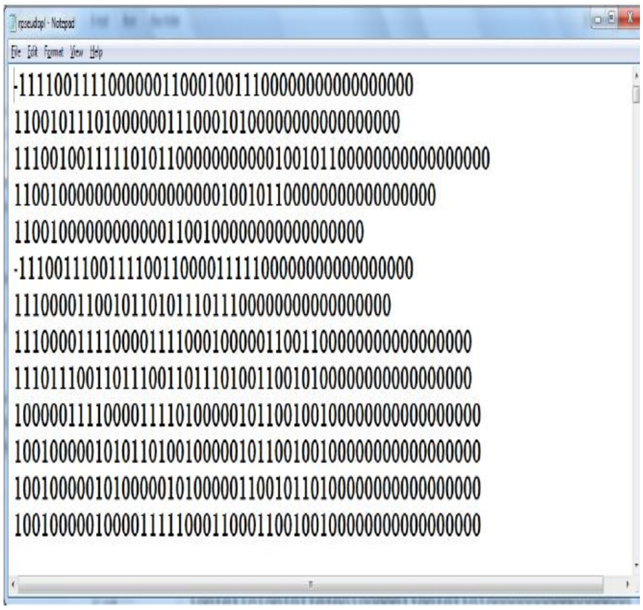
Figure 5: Binary form of Pseudo plaintext



```
-1001010011100101000000100001001
1110111110101101010001001010101001
111111111101000100110001111000011100100101 01
11111100011110011110110100001100000110 1
1000101011000001001001011000 10001
-1000000101011111101111001010000111111 11
11000010101110011100111111101110101 01
1000100101011111001100010001010010110010100 1101
110100011000000000100101000111001010110010100001
10011001000001110110101011011110010010101100010 1
1100101100111000111100011101001001000100110010 01
10000010001000011111010001111010000001111110010 01
10101100100001111011100010001010011010001001001 01
```

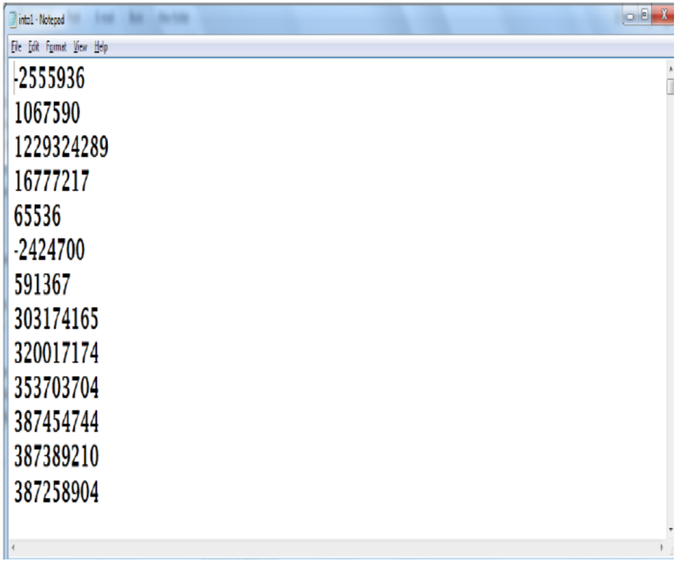
Figure 6: Set of ciphers.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication



```
-1111001111000000110001001110000000000000000000
110010111010000001110001010000000000000000000
1110010011111010110000000000010010110000000000000000
11001000000000000000000010010110000000000000000
1100100000000000011001000000000000000000
-111001110011110011000011111000000000000000000
1110000110010110101110111000000000000000000
11100001111000011110001000001100110000000000000000
1110111001101110011011101001100101000000000000000000
1000001111000011110100000101100100100000000000000000
1001000001010110100100000101100100100000000000000000
1001000001010000010100000110010110100000000000000000
1001000001000011111000110001100100100000000000000000
```

Figure 7: Recovered from ciphers.



```
-2555936
1067590
1229324289
16777217
65536
-2424700
591367
303174165
320017174
353703704
387454744
387389210
387258904
```

Figure 8: Original plain text parts in integer form.

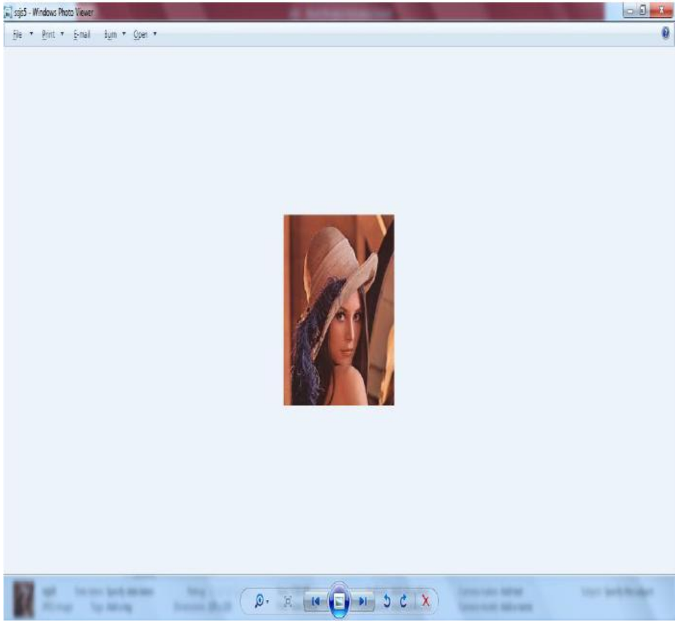


Figure 9: Decrypted original file.

VI. Conclusion and Future work

Cloud computing is an inevitable technology from startups to renowned organizations. Proper security measures have to be taken when data is moved to cloud. In this paper, a novel Invertible Non-linear Function based Cryptographic System (INFCS) is proposed. The proposed scheme stores each encrypted data in different data centers. Hence, an adversary has to know as many locations of data centers as the number of encrypted data, which is not easy. In the worst case scenario, even if all ciphers are obtained by the adversary, original plain text cannot be obtained, since a cloud storage or cloud storages contain(s) encrypted data only. An authorized user can alone derive original data from encrypted data with the key and knowledge of $g^{-1}(x)$. Thus the proposed technique is efficient and faster. In future, the proposed INFCS will be applied for encrypting video/audio files.

REFERENCES

1. Seny Kamara and Kristin Lauter, "Cryptographic Cloud Storage", Financial Cryptography and Data Security, Volume 6054, 2010, pp 136-149.
2. R. Arokia Paul Rajan and S. Shanmugapriyaa, "Evolution of Cloud Storage as Cloud Computing Infrastructure Service", IOSR Journal of Computer Engineering (IOSRJCE) ISSN : 2278-0661 Volume 1, Issue 1, 2012, pp 38-45.
3. G. Murugaboopathi, C.Chandravathy and P. Vinoth Kumar, " Study on Cloud Computing and Security Approaches", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume 3, Issue 1, 201, pp 504-506.
4. G. Clarke, Microsoft's Azure Cloud Suffers First Crash, The Register, 2009, [online] http://www.theregister.co.uk/
5. Hassan Takabi , James B.D. Joshi and Gail Joon Ahn, "Cloud Computing Security and Privacy Challenges in Cloud Computing Environments ", COPUBLISHED BY THE IEEE COMPUTER AND RELIABILITY SOCIETIES, 2010, pp. 1540-7993.
6. Australian government, department of defense, "Cloud Computing Security Considerations", CYBER SECURITY OPERATIONS CENTRE, 2012, pp 1-18.
7. Richard Chow, Philippe Golle, Markus Jakobsson, Elaine Shi, Jessica Staddon, Ryusuke Masuoka, and Jesus Molina, "Controlling Data in the Cloud: Outsourcing Computation without Outsourcing Control", In Proceedings of the 2009 ACM workshop on Cloud computing security, CCSW "09, New York, NY, USA, 2009. ACM, pp 85–90.
8. Neha Jain and Gurpreet Kaur, "Implementing DES algorithm in Cloud for Data Security", VSRD International Journal of CS & IT Volume 2 Issue 4, 2012, pp. 316-321.
9. Somdip Dey "SD-AREE: An Advanced Modified Ceaser Cipher Method to Exclude Repetition from a Message", International Journal of Information & Network Security (IJINS). Volume 1, Issue 2, 2012, pp. 67-76.
10. Amit joshi and Bhavesh Joshi, "A Randomized Approach for Cryptography", International Conference on Emerging Trends in Network and Computer Communications (ETNCC), 2011, pp. 293-296.
11. Quist-Aphetsi kester, "A Hybrid cryptosystem based on Vigenere cipher and columnar transposition cipher", International Journal of Advanced Technology & Engineering Research, Volume 3, Issue 1, 2013, pp. 141-147.
12. Fadhil Salman Abed, "A Proposed Method of Information hiding based on Hybrid Cryptography and Seganography", International Journal of Application or Innovation in Engineering & Management, Volume 2, Issue 4, 2013, pp 530-539.
13. Ajit Singh, Aarti Nandal and Swati Malik, "Implementation of Ceaser Cipher with Rail Fence for enhancing data Security", International Journal of Advanced research in Computer Science and Software Engineering. Volume 2, Issue 12, 2012, pp 78 -82.

14. Wu, Jiehong, Ilia Detchenkov, and Yang Cao, "A study on the power consumption of using cryptography algorithms in mobile devices", Software Engineering and Service Science (ICSESS), 2016 7th IEEE International Conference on. IEEE, 2016, pp 957-959.
15. B. Wang, M. Li, S.S. Chow and H. Li, “A tale of two clouds: computing on data encrypted under multiple keys”, IEEE Conference on Communications and Network Security (CNS), IEEE, 2014, pp. 337–345.
16. Bhandari, Akshita, Ashutosh Gupta, and Debasis Das "Secure algorithm for cloud computing and its applications", Cloud System and Big Data Engineering (Confluence), 6th International Conference-Cloud Systems and Big Data Engineering IEEE, 2016, pp 188-192.
17. G.Ramachandra, M. Iftikhar and F. A. Khan, “A Comprehensive Survey on Security in Cloud Computing”, The 3rd International Workshop on Cyber Security and Digital Investigation, Procedia Computer Science 110, 2017, pp 465-472.
18. A.Nithya B. ,Ramakrishnan ,Resul Das “A Novel Approach for Data Privacy UsingAttribute Based Scheme Algorithm for Cloud Computing” International Journal of Computer Networks and Applications, pp . 70-77
19. Reham M. Abobeah, Mohamed M. Ezz, Hany M. Harb “Public-Key Cryptography Techniques Evaluation” International Journal of Computer Networks and Applications, pp . 64-75