Implementation of Security Algorithm for Data Security in Cloud Computing

Mohd Amir Siddiqui, M Akheela Khanum

Abstract*: Cloud technology offers on-demand access/allocation of resources (such as storage, computing) as guaranteed and reliable services. Cloud users valuable and confidential data is stored on a virtual server that may be shared by multiple users. This stored data is prone to attacks (access or alteration) for malicious intentions by an unauthorized users or program. So, security of data is a big challenge for cloud computing. However, such an instance can be prevented if the stored data is encrypted. Existing encryptions employ algorithms that are towards the end of life, thanks to ever improving computing technologies. This paper proposes an encryption algorithm which is efficient and optimized for data security in cloud computing. The proposed Algorithm performs multiple rounds of encryption operations based on the length of the encryption key, harnessing the power of parallel processing to ensure better performance. The main features of this algorithm are that it uses a larger key and is designed in a way that fragment of code can also be executed in parallel.

***Index Terms*: Cloud Security, Data Security, Encryption, Parallel Processing.**

I. INTRODUCTION

Cloud computing is a type of computing which provides the facility to use resources available on cloud system, or we can say that it is a model where resources are retrieved through network. It allows user to use technology enabled services over the internet[1]. Cloud technology works as abstraction layer between cloud user and resources enabling them to use services without getting into the management operations. Cloud service provider employs a number of virtual machines to efficiently share and allocate the resources among its users on the basis of their respective demands. In cloud computing several users or organizations store their data on a single virtual server, sometimes multiple operating systems are executed on a single virtual server. This concept is termed as multi tenancy, which opens avenues of additional threats[2]. So, there is a need of higher level of security especially in public cloud system. One of the major benefits offered by Cloud Technology is that physical location of a device is transparent to the user[3]. The user only need to interact with an interface and the actual device can be placed anywhere in the world, this also helps in disaster management. Also, a single resource or device can be shared among many users. This helps in achieving better utilization. A cloud service is considered as more reliable as the cloud service provider maintains backups. Cloud technology also supports scalability and elasticity as cloud services use a pool of resources[2]. Centrally managed data and the insulation between programs and data make it more secure. A cloud system can also be self managed as all the services and allocations can be performed by a software tool as and when a user makes a request[3]. Cloud computing technology brings many challenges along with its remarkable benefits. As stated by many authors, technical writers, and concluded by surveys of prominent media houses that security is still a major concern in context of cloud technology[1], [3]–[6]. Privacy, Compliance, Interoperability, Lock-in are other major concerns that are parts of the picture[1]. Frequent events of security and privacy breaches, make them top the list. However, the threat of compliance and interoperability is decreasing slowly and gradually. Lock-in still haunt users from handing over their precious data to the clouds. Apart from these, SLA level, computational and data level challenges are also there [2]. An online comprehensive survey [4] conducted by Crowd Research Partners states in Cloud Security report that cloud security is a big concern for cloud users and security professionals. However, a RightScale report states that “As companies become more experienced with cloud, the top challenge shifts. Security is the largest issue among cloud beginners, while cost becomes a bigger challenge for intermediate and advanced users” [5]. As data is gaining importance in this digital era, it is attracting the interest of data thieves and hackers and therefore becomes a prime concern for security professionals. In cloud technology, physical storage resources are shared among users which attract the threat of unauthorized access to data. Also, the user always stay in fear of data loss and data leak, as the data is stored away from him at an unknown distant location[7]. This paper describe proposed an encryption algorithm that suits the specific needs of cloud computing by harnessing the parallel processing to offer better response time and throughput. The algorithm described in this paper will serve as a base to formulate security algorithm for application specific needs of the users.

II. LITERATURE REVIEW

This section reviews various existing algorithms that were developed to provide data security using encryption. Existing algorithms are reviewed on their known merits and demerits.

Manuscript published on 30 March 2019.
***Correspondence Author(s)**

Mohd Amir Siddiqui, CSE Department, Integral University, Lucknow, India.
Dr. M. Akheela Khanum, CSE Department, Integral University, Lucknow, India.

A. Data Encryption Standard (DES)

A popular block cipher (developed by IBM in 1972) uses 56 bit key to encrypt a 64 bit blocks at a time. However, weak key makes it prone to attacks. It supports OFB, ECB, CFB and CBC modes. In 1999, it was cracked using "Deep Crack" supercomputer with the help of lakhs of distributed computers over the internet, in less than a day i.e. twenty two hours[8].

B. International Data Encryption Algorithm (IDEA)

IDEA (first published in 1991) uses a 128 bit key to operate eight and a half rounds of operations to encipher a 64 bit block. Its shortcoming lies the key schedule which lead to weak keys for a which fixes are proposed such as XORing key with a constant, but by 2002, more and larger classes of weak keys were found [9].

C. Triple-DES

Triple-DES (first described in 1998), as the name suggests, operates DES algorithm, thrice (Encryption, Decryption and again Encryption) over each block of data to get the final cipher data with a 112/168 bit key and 64 bit block. This also couldn't prove to be worthy for longer times due to the weaknesses of the base algorithm i.e. DES and ever improving computing abilities[10].

D. Advanced Encryption Standard (AES)

AES (Advanced Encryption Standard, developed in 1998) encrypts a 128 bit block of data with any one of 128, 192 and 256 bit key[11]. It first breaks 128 bit in 4 blocks of 64 bit and these blocks are then operated with the transformations based on permutations and substitutions. The number of transformations depends on the size of keys. This technique works well when implemented on hardware also[12].

E. Blowfish and Twofish

Blowfish (first published in 1993) is a symmetric fiestal block cipher that uses a 32 to 448 bit key to encrypt a 64 bit block data. The advantage of having a range of key sizes makes it a preferred choice for personal as well as professional use. However, the size of data block of plain text makes it a victim of birthday attacks [13]. Twofish comes as a replacement for the shortcomings of blowfish, having a 128 bit data block. This algorithms quality lies in the use of complex key schedule and Key Dependent Substitution boxes. Another reason for being a popular choice is that it is free to be used by anyone[14].

The review of algorithms mentioned above clearly shows that as of current standings, only AES and Twofish, still have capabilities to survive cryptanalysts or attackers for significant time in future. Also, only these algorithms are suggested by cryptographers for standard use [12], [14], [15]. Though Twofish has not got much popularity, AES serves as a standard encryption algorithm for a large section of security needs. Therefore, the proposed algorithm is compared with these two algorithms only.

III. PROPOSED ALGORITHM

This section describes the proposed algorithm "Enhanced Encryption Algorithm (EEA)". The algorithm operates with 512-bit plaintext and cipher text blocks at a time and is operated by a 512-bit key. The algorithm structure has been chosen such that, the encryption process is identical to the decryption process.

A. Key Generation

A random initial key of 512 bit is generated by the system. This key is then partitioned in four 128 bit sub blocks. These four blocks will be used for initiation in encryption process. Further the key for next steps is generated using a sequence of operations on existing key as done in AES.

B. Encryption:

The encryption is performed in multiple rounds of the following steps:

Step 1: A 512 bit plaintext block is taken and partitioned into four 128 bit sub blocks. For each sub block of message and corresponding block of key, further steps are followed.

Step 2: Key block is added in Message block

Step 3: This step consists of 3 sub-steps:

a. *Byte Replacement step*: Each byte of output of previous is replaced by another using mapping scheme/ replacement table.

b. *Mirror Swap step*: For this step, 128 bits from the output of previous step are arranged as a 2D array. Then, mirror image of this 2D array is taken.

c. *Rotation step*: Bytes in each word of the 2D array generated after previous step is right shifted R times, where R= (no. of step) mod (size of block).

d. *Add stepKey*: stepKey generated by a separate process using the initial key/ previous stepKey is then added to the output of previous step. Skip this step on tenth iteration.

Step 4: Repeat previous steps ten times; on completion we get our cipher text block.

IV. IMPLEMENTATION

The proposed encryption algorithm was developed and implemented in java and deployed on a cloud licensed from Digital Ocean [16]. The experimental environment runs on droplet[17] with configuration of 4vCPU (Intel Xeon Processor), 8 GB Memory and 160 GB SSD. Implementation setup is developed using primarily angular js, mysql and java. Implementation of AES is taken from java crypto and security libraries[18] while the implementation of Twofish and

Table I: Details of files used for implementation and testing purposes

S. No.	File Size	File Type
1.	364 KB	Powerpoint presentation
2.	1183 KB	PDF Document
3.	5126 KB	PDF Document
4.	10.12 MB	MP3 file
5.	24.4 MB	MP4 file
6.	50.7 MB	MP4 file

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Proposed algorithm is performed with the help of java crypto and security libraries.

The values of comparison parameters were stored in the database for six files (table I) with different file formats and file sizes. All six files were encrypted using encryption algorithms to evaluate them on comparison parameters.

V. RESULT AND DISCUSSION

The encryption algorithm was evaluated using following parameters:

A. Encryption time

The encryption time is noted for the all three algorithms and a graph is plotted (shown in figure 1) and the corresponding values are given in table II. The graph shows that EEA is performing significantly better than the other two algorithms.

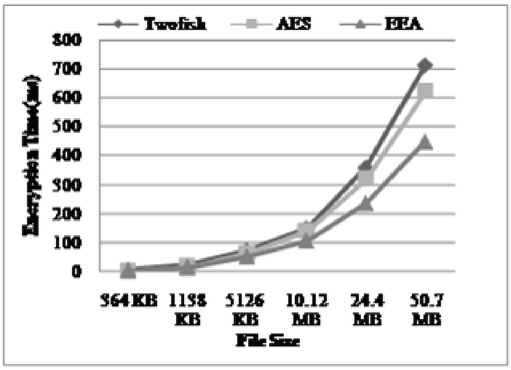

Figure 1: Graph showing Encryption Time of different files for each algorithm

Table II: Table of Encryption times of different files for each algorithm

Encryption Time (in ms) vs File Size	**364 KB**	**1138 KB**	**5126 KB**	**10.12 MB**	**24.4 MB**	**50.7 MB**
Twofish	3	18	71	148	359	715
AES	2	15	62	138	321	625
EEA	2	13	50	105	235	449

B. Decryption time

The decryption time is noted for the three algorithms and a graph is plotted (shown in figure 2) and the corresponding values are given in table III. In the graph, we can see notable difference between EEA and the other two, in which EEA, outperforms the rest.

Table III: Table of Encryption Times of different files for each algorithm

Decryption Time (in ms) vs File Size	**364 KB**	**1138 KB**	**5126 KB**	**10 .12 MB**	**24.4 MB**	**50.7 MB**
Twofish	3	17	68	151	348	708
AES	2	15	64	135	318	628
EEA	2	12	51	103	221	446

C. Entropy[19]:

It signifies the extent of randomness in the data. Every meaningful data shows some relation among the data. This relation may disclose useful information regarding secret message. Therefore, it is desired that the randomness of any encrypted message should be high. We have calculated the value of Entropy of Encrypted file using 16-bit Entropy (Shanon's formulae[20]) calculator, an open-source tool[21].

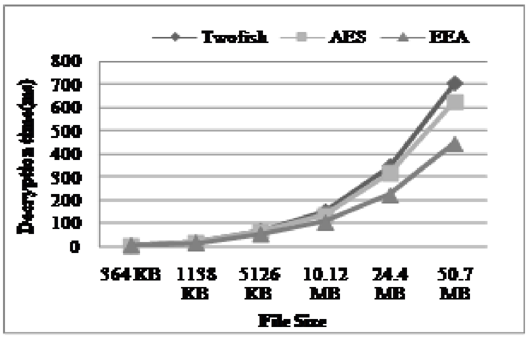

Figure 2: Graph showing Encryption Time of different files for each algorithm

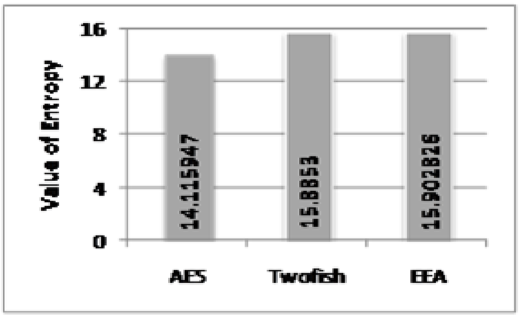

Figure 3: Average Entropy value of encryption files for the three algorithms

Higher Value of Entropy prevents statistical attacks on cipher. Average Entropy value is calculated from 15 encryptions for each of the three algorithms, and is shown in bar graph in fig. 3. The entropy value of EEA is around max value, which shows better degree of randomness and in turn, quality of encryption.

D. Avalanche Effect[22]:

This effect states that a very small change in the input will lead to a very big change in the output. It is desirable that even a change of one input or key bit results in changing more than half output bits. For a good algorithm, value of avalanche effect should be greater than 50%. Figure 4 shows the avalanche effects value for each algorithm.

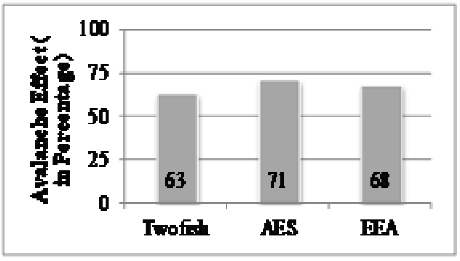

Figure 4: Avalanche effect value of encryption files

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Table IV: Comparison of general factor of AES, TwoFish and EEA

Factors	**AES[11]**	**Twofish**[14]	**EEA**
Development Year	2001	1998	2018
Key Length	128, 196, 256	128, 196, 256	512 bit (Can be expanded)
Block Size	128-bit	128-bit	512 bit
Rounds	10, 12, 14 (Depends on Key)	16	10 (Depends on key)
Encryption Speed	Fast	Slightly slower than AES	Faster than AES
Quality of Encryption	Very High	Very High	Very High
Structure	Substitution–permutation network	Feistel network	Substitution–permutation network

E. Other General Factors:

Table IV shows comparison of various other general properties of the three algorithms

F. Key Space of EEA:

EEA uses a key space of 512 bits, whereas Twofish and AES uses 128, 192, 256 bit keys. Generally, 128 bit key is used for AES and Twofish. Even if, both existing algorithms use 256 bit key, then also EEA key space is 2^{256} times bigger. The time taken to scan a 512-bit key space in order find the possible key can roughly be as follows:

- As of June 2018, the fastest supercomputer on the TOP500 supercomputer list is the Summit[23], in the United States, with a LINPACK benchmark score of 122.3 PFLOPS.

  122.3 PFLOPS = $122.3*10^{15}$ FLOPS$<=10^{18}$ FLOPS
- Now if we take 10000 supercomputers, which are 100 times faster than Summit and assume that processing a key takes no more than 1 FLOP of time. Then, the combine processing system will process 10^{22} keys/second, still it will roughly, take $12*(10^3)^{41}$ years (= $12*10^{123}$ years) to break cipher text.

VI. CONCLUSION

Privacy and security of data stored on cloud is most important for the users of cloud infrastructure. This paper review existing security methods used to secure users data in clouds. This paper has proposed an encryption algorithm “Enhanced Encryption Algorithm (EEA)” to encrypt uploaded files on cloud storage. The discussion shows that EEA provides a stronger encryption. Also, It takes up to 28% lesser time for encryption than AES and up to 37% lesser time than Twofish. Avalache effect and entropy value proves the quality of cipher text produced by the proposed algorithm is comparable to industry standards. As described in previous sections, key strength is practically impossible to be broken in near future. Further, the trend of encryption time shows EEA becomes more beneficial as the file size increases, which makes it favorable for Cloud usage. However, parallel execution of code fragment incurs storage and computing overhead, which is a very small price for better performance.

However, studies can be performed for key expansion, key storage, key management, cryptanalysis and effect of various attacks.

ACKNOWLEDGMENT

We would like to extend gratitude to Faculty of Doctoral Studies and Research, Integral University for their support. (Manuscript Comm. No.: IU/R&D/2019-MCN000564).

REFERENCES

1. N. Subramanian and A. Jeyaraj, “Recent security challenges in cloud computing,” Comput. Electr. Eng., vol. 71, pp. 28–42, 2018.
2. H. Mezni, S. Aridhi, and A. Hadjali, “The uncertain cloud: State of the art and research challenges,” Int. J. Approx. Reason., vol. 103, pp. 139–151, 2018
3. F. Shahzad, “State-of-the-art Survey on Cloud Computing Security Challenges, Approaches and Solutions,” Procedia Comput. Sci., vol. 37, pp. 357–362, 2014.
4. C. I. Schulze, Holger, CEO, “Cloud Security Report 2018,” 2018.
5. RightScale Inc., “RightScale 2018 State of The Cloud Report,” 2018.
6. C. Stergiou, K. E. Psannis, B. B. Gupta, and Y. Ishibashi, “Security, privacy & efficiency of sustainable Cloud Computing for Big Data & IoT,” Sustain. Comput. Informatics Syst., vol. 19, pp. 174–184, 2018.
7. P. R. Kumar, P. H. Raj, and P. Jelciana, “Exploring Data Security Issues and Solutions in Cloud Computing,” Procedia Comput. Sci., vol. 125, pp. 691–697, 2018.
8. SANS, “Interested in learning SANS Institute InfoSec Reading Room In tu , A ll r igh,” Worm Propag. Countermeas., p. 36, 2004.
9. A. Biryukov et al., “New Weak-Key Classes of IDEA,” Lect. Notes Comput. Sci., pp. 315–326, 2002.
10. E. Barker, W. Barker, and W. Burr, “Recommendation for Key Management,” NIST Spec. Publ. 800-57, pp. 1–142, 2007.
11. NIST, “FIPS PUB 197: Specfication for the Advanced Encryption Standard (AES),” 2001.
12. A. Biryukov and D. Khovratovich, “Related-Key Cryptanalysis of the Full AES-192 and AES-256,” in Advances in Cryptology -- ASIACRYPT 2009, 2009, pp. 1–18.
13. K. Bhargavan, “On the Practical (In-) Security of 64-bit Block Ciphers Collision Attacks on HTTP over TLS and OpenVPN,” Ccs 2016, pp. 456–467, 2016.
14. B. Schneier, J. Kelsey, D. Whiting, D. Wagner, and C. Hall, “Twofish : A 128-Bit Block Cipher,” Current, vol. 21, no. 1, pp. 1–27, 1998.
15. J. Hou, L. Qu, and W. Shi, “A survey on internet of things security from data perspectives,” Comput. Networks, vol. 148, pp. 295–306, 2019.
16. “Digital Ocean.” [Online]. Available: https://www.digitalocean.com/. [Accessed: 12-Dec-2018].
17. A. Bodaghkhani, Y. S. Muzychka, and B. Colbourne, “An analytical model of final average droplet size prediction of wave spray cloud,” Int. J. Heat Fluid Flow, vol. 74, pp. 110–117, 2018.
18. Oracle Network Technology, “JAVA Cryptography Architecture.” [Online]. Available: https://docs.oracle.com/javase/8/docs/technotes/guides/security/StandardNames.html#Cipher. [Accessed: 09-Jan-2019].
19. S. Kumar, M. Kumar, R. Budhiraja, M. K. Das, and S. Singh, “A cryptographic model for better information security,” J. Inf. Secur. Appl., vol. 43, pp. 123–138, 2018.

20. P. Pathria, R. K.; Beale, Statistical Mathematics, Third. Academic Press, 2011.
21. Server Test, “Entropy and Randomness Online Tester.” [Online]. Available: https://servertest.online/entropy. [Accessed: 12-Nov-2018].
22. A. F. Webster and S. E. Tavares, “On the Design of S-Boxes,” in Advances in Cryptology --- CRYPTO ’85 Proceedings, 1986, pp. 523–534.
23. J. Dongarra, E. Strohmaier, H. Simon, M. Meuer, and H. Meuer, “TOP500 List - June 2018,” Top 500 list. [Online]. Available: https://www.top500.org/list/2018/06/. [Accessed: 13-Nov-2018].

AUTHORS PROFILE

Mohd Amir Siddiqui is an academician and research scholar having 7 years of experience in academics and research. He has more than 8 research publications to his credit

Dr. M. Akheela Khanum is an academician having 16 years of experience in academics and research. She worked in several national and international institutions of repute on various positions. She has more than 40 research publications to her credit. She is a reviewer and editorial board member of several international journals and conferences