Ensuring Data Security in Cloud Storage

Shubham Singh and Akhilendra Pratap Singh

***Abstract*—Cloud computing is one of the most prominent storage and computing platform being researched nowadays. It is amongst the most popular networks in the world and is seen as next generation of computing technology. It provides various services to its users. Among them, one of the salient service offered is cloud storage. But the major concern while using this service is the integrity and privacy of stored data. Users require their data to be safe and private from any modification or unauthorized access. Thus security becomes an indispensable part of the data stored on cloud. This paper proposes a way to maintain data privacy and integrity and guarantees that no one except the authorized user can access the data.**

***Index Terms*—Cloud computing, integrity, privacy, security.**

I. Introduction

Cloud computing is defined as a network based computing which provides shared processing data & resources to its user when required. As per NIST, it is a model which enables pervasive, on-demand access to shared resources that can be rapidly provisioned & released with minimal management effort [1]. It provides a user various capabilities for storing their data at the cloud server & processing it when required. The availability of cheap storage devices & computers, high capacity networks, service oriented architecture etc. are behind the growth of this technology.

Cloud computing offers various advantages like better hardware utilization, reduced cost, high scalability etc. [2] to both the cloud service provider and its users, and is now a highly demanded service. But despite of such advantages, organizations prefer to adopt the methodologies that are already successful and used by other organizations. This is because of the risks associated with cloud computing.

There are some privacy concerns in the services provided by the cloud [3]. The cloud service provider can access the stored data any time and can modify or delete it. It can share the stored information with third parties, if necessary, as permitted in their privacy policy. A solution to protect the confidentiality of stored data is the way in which user stores data. User can encrypt data before storing it in the cloud to prevent unauthorized access. Also, in a cloud provider platform, used by a large number of users, the data belonging to different users may reside on same data server which may lead to information leakage when a user's information is given to other [4]. There is also a problem regarding the ownership of the stored data i.e. if a user stores his data on cloud, who the owner of stored data is?

Manuscript received May 20, 2018; revised July 8, 2018.

The authors are with the National Institute of Technology Meghalaya, Shillong, India (e-mail: ShubhamSinghCMR@nitm.ac.in, akhilendra.singh@nitm.ac.in).

II. Related Work

Confidentiality, Integrity and Availability are the key attributes for any data and these problems cannot be solved by a single security method. Liu discusses about the traditional technologies and capabilities of cloud and the newer technologies which must be used for better security and privacy of data [5]. The importance of security of data stored in the cloud has been emphasized many a times, along with the measures needed to secure the data in the cloud. In [6], AES algorithm is used to provide security to the end users of Cloud. An encryption key is used to encrypt the user files to secure the content of the files. The encrypted files can be transferred over the network without worrying about the files getting accessed by an unauthorized user. Al-Jaberi et al. main focus is integrity verification along with privacy preservation using algorithms and protocols for cloud stored data where Amazon S3 was used as the cloud storage provider [7]. RSA partial homomorphic & MD5 algorithm are used in the proposed model [8]. RSA algorithm is used for securing data and MD5 algorithm is used for integrity verification. The author enhanced the cloud security mechanism using AES 128 bit, 192 bit & 256 bit key encryption to secure the data in the cloud, depending on the size of the files [9]. In [10], for securely transmitting & storing data in the cloud, encryption & decryption, both are done at clients end using a single key. Mahalle et al. used data encryption using hybrid encryption algorithm to ensure data security such that even the administrator of the cloud server does not have access to the private data of the user [11]. The keys are generated on the basis of system time thereby making the overall system more secure. In [12], the author proposed a new security model with double authentication mechanism to be implemented to restrict unauthorized persons from getting the control of user's data. The user can also select the encryption techniques he wants to use.

People are afraid of sharing their data with an untrusted cloud service provider. The proposed model uses encryption and obfuscation techniques to guarantee higher level of security and confidentiality of user's data [13]. Prasad et al. proposed the implementation of High Security Password to ensure the security of user's data [14]. Any modification in data will be done only if the user provides the password sent to his mobile. This is more secure than the traditional security mechanism of using a single password for the whole profile and tasks related to that account. The author has proposed secure sharing functionality to be added for cloud service providers using cryptographic algorithms like AES and RSA, by associating particular permission decided by owner with generated keys that can be used to access a resource [15]. The resource can only be accessed if the key used by the user has

doi: 10.18178/ijmlc.2018.8.4.716

the required permissions associated with it. The proposed model uses a combination of authentication technique (digital signature) and key exchange algorithm (Diffie Hellman) blended with an encryption algorithm (AES) referred to as "Three-way mechanism" to be used to protect confidentiality of data stored in the cloud [16]. In [17], different security issues arising from the usage of cloud services are discussed. The author has proposed if one wants to take the cloud computing to the next level, the security capabilities need to be strengthened. Elliptic curve cryptography encryption technique is used for securing the data present in the cloud so that only authenticated user can access the data [18]. Any breach at the cloud would not affect the security of stored data as the data is in encrypted form. The author has covered all the cloud security techniques & countermeasures along with the attacks possible against cloud based services and the countermeasures to thwart these attacks [19]. Mohamed et al. implements a software on Amazon EC2 Micro instance to enhance data security in [20]. Also, the author gives three suggestions for Amazon EC2 users: algorithms like blowfish or DES can be used for better performance, AES must be used if better security of user's data is needed, and AES suits Amazon EC2 by providing better performance and security.

III. Problem Statement

There are several security issues associated with cloud storage like loss of data, loss of control, invalid storage etc. [21]. They are a major obstruction in adopting cloud computing services as the confidentiality and integrity of stored data is very important. Once a user stores his data on cloud, there must be some assurance that the data can be accessed only by the authorized user and will remain confidential. But due to some internal errors or malicious changes, user's data might be exposed to unauthorized individuals or given to authorized users with integrity being compromised.

This paper put forwards a new approach which ensures the privacy along with the integrity of the data stored at cloud server.

IV. Preliminaries

A way to provide security to data is cryptography which uses logic and mathematics to serve the purpose. Advanced Encryption Standard (AES) is such a cryptographic technique, a symmetric block cipher which performs all of its computations on bytes [22]. The number of rounds involved are variable & depends on key length. AES is used by a number of organizations as it is more secure than other algorithms and is faster in both hardware and software. Encryption and decryption of data is easy and gives good performance. It supports large key sizes of 192 and 256 bits along with its 128 bit form for heavy encryption purposes. AES is considered invulnerable to all security attacks except brute force. When its most secure 256 bit key is used, it will take a large number of years to guess the key by brute force attack and hence it is almost unbreakable [23].

Encryption itself is not sufficient for ensuring data's integrity. When a user stores his data on cloud, he needs an assurance that it must remain intact. Therefore, Secure Hash Algorithm 2 (SHA-2) is used, which is designed to protect the integrity of data by detecting any modifications and warns the owner about it. It is a set of cryptographic hash functions which run on user's data by comparing the computed hash value to a previously known hash value. Based on the digest length, there are six hash functions in SHA family. Their shift amount is different but they have identical structure which differs in number of rounds [24]. SHA-2 is a stronger hashing algorithm in comparison to other and is more suitable for verifying the integrity of data as it is less susceptible to most cryptanalytic attacks.

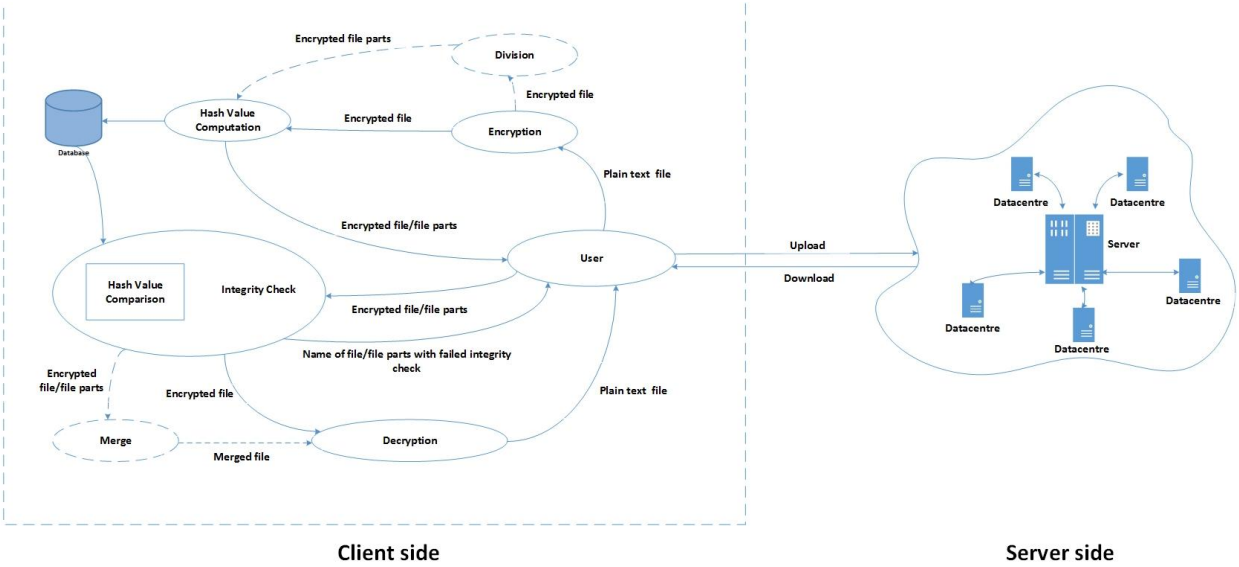

Fig. 1. Proposed model.

V. Proposed Scheme

Fig. 1 shows the architecture of proposed scheme. It supports client side encryption so that data is transmitted securely over the network. For it, AES algorithm is used as it has the computational efficiency and speed to encrypt large data. The proposed algorithms are as follows:

ALGORITHM #1: AES Encryption
F: Selected file
F_NAME: File name
EF: Encrypted file
IVAL: Hash value of encrypted file
N: Number of encrypted file parts
EF_PART_NUM: Name of a part of encrypted file
K: Encryption key

1. *start*
2. *F=select_file();*
3. *K=enter_key();*
4. *EF=AES_encrypt(F,K);*
5. *IVAL=find_SHA_val(EF);*
6. *insert_db(F_NAME,IVAL);*
7. *if (split file in parts) then*
   a. *file_split(EF);*
   b. *for NUM=1 to N do*
      i. *IVAL=find_SHA_val(EF_PART_NUM);*
      ii. *insert_db(EF_PART_NUM,IVAL);*
      iii. *output EF_PART_NUM ;*
      iv. *NUM++;*
      v. *end for*
   c. *goto 9;*
8. *else*
   a. *output EF;*
   b. *goto 9;*
   c. *end if*
9. *end*

ALGORITHM #2: File Division
F: Selected file
FS: Selected file size
PS: Size of a part of the file
RL: Total number of bytes to be read from the selected file
READ: Stores the bytes read from the selected file
IVAL: Hash value of encrypted file part
F_PART_NUM: New file part

1. *start*
2. *F=select_file();*
3. *PS=input_partSize();*
4. *FS=F.length();*
5. *if FS>0 then*
   a. *goto 7;*
6. *else*
   a. *output "Blank file";*
   b. *goto 8;*
   c. *end if*
7. *while FS > 0 do*
   a. *if FS <= PS then*
      i. *RL=FS;*
   b. *else*
      i. *RL=PS;*
      ii. *end if*
   c. *READ=byte_read(F,0,RL);*
   d. *FS-=RL;*
   e. *F_PART_NUM=create_file();*
   f. *write_file(F_PART_NUM,READ);*
   g. *IVAL=find_SHA_val(F_PART_NUM);*
   h. *insert_db(F_PART_NUM,IVAL);*
8. *end*

ALGORITHM #3: Cloud Server Process

1. *start*
2. *client_login();*
3. *if (!client_login_success()) then*
   a. *output "Incorrect credentials";*
   b. *goto 2;*
4. *else*
   a. *goto 5;*
   b. *end if*
5. *if (file_upload) then*
   a. *choose_server();*
   b. *select_file();*
   c. *upload_server();*
   d. *goto 5;*
6. *else if (file_download) then*
   a. *choose_file();*
   b. *download_file();*
   c. *goto 5;*
7. *else if (logout())then*
   a. *goto 8;*
   b. *end if*
8. *end*

ALGORITHM #4: AES Decryption
F: Selected encrypted file
MF: Merged file
Fi: Temporary file
K: Decryption key
DF: Decrypted file
N: Number of encrypted file parts
EF_PART_NUM: Name of a part of encrypted file

1. *start*
2. *K=enter_key();*
3. *if (file divided in parts) then*
   a. *MF=file_merge();*
   b. *Fi=MF;*
   c. *goto 5;*
4. *else*
   a. *Fi=F;*
   b. *goto 5;*
   c. *end if*
5. *if (check_integrity(Fi)) then*
   a. *DF=AES_decrypt(Fi,K);*
   b. *output DF;*
   c. *goto 7;*
6. *else*
   a. *if (file divided in parts) then*
      i. *output "Modified file parts";*
      ii. *for NUM=1 to N do*
         1. *if(!check_integrity(EF_PART_NUM)) then*
            a. *output EF_PART_NUM;*
            b. *NUM++;*
         2. *else*
            a. *NUM++;*
            b. *end if*
         3. *end for*
      iii. *goto 7;*
   b. *else*
      i. *output "File modified";*
      ii. *goto 7;*
      iii. *end if*
   c. *end if*
7. *end*

ALGORITHM #5: File Merge
F: File name
L: Array list
N: Number of file parts
F_PART_NUM: A part of the file
F_M: Merged file
READ: Stores the bytes read from the selected file

1. *start*
2. *F= enter_fileName();*
3. *L=create_newList();*
4. *for NUM=1 to N do*
   a. *L.add(F_PART_NUM);*
   b. *end for*
5. *F_M= create_file();*
6. *for each F_PART_NUM in L do*
   a. *READ=read(F_PART_NUM,0,F_PART_NUM.length())*
   b. *write(F_M,READ);*
   c. *end for*
7. *end*

VI. Result and Performance Analysis

The proposed model is successfully implemented using java crypto package, javax.crypto, jre v1.8.0 and PHP v5.4.38 on windows machine with Intel core i3 processor TM-4005U running at 1.70 GHz and 4G memory.

As confidentiality of users data is most essential, therefore to ensure it various cryptographic algorithms are there which can be used to protect the data from unauthorized access. Although security is the most important factor, there are some other factors which must be taken into consideration while choosing an algorithm for encrypting the data, most important among which is the time taken for data encryption and decryption. A comparison between some of the cryptographic algorithms is provided in [25] on the basis of time taken to encrypt a file. A text file of 50 MB in size is chosen as a sample whose encryption time by different algorithms is as depicted in Fig. 2. From the graph it is clear that AES algorithm takes the minimum execution time for data encryption and decryption when compared with other cryptographic algorithms. Therefore AES algorithm is the best option when execution time is taken into consideration.

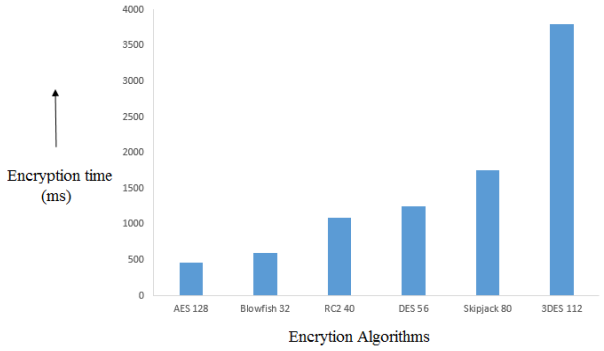

Fig. 2. Encryption algorithm vs. time.

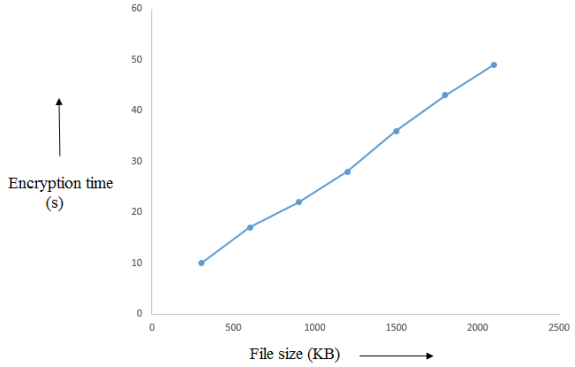

Fig. 3. File size vs. encryption time.

The performance of an algorithm on same data of different sizes is also important. Fig. 3 describes the time taken by AES algorithm to encrypt a data file of different sizes [25]. It is clear from the graph that as the data size increases, encryption time of AES algorithm increases and vice versa. Thus, the time taken by AES algorithm to encrypt a file is directly proportional to the file size. It means that a file having large size will take more encryption time then the file having less size.

VII. Discussion

The proposed model enhances the security of data. It is useful for data whose protection is required because of the source or the nature of individuals involved, or when it is stored in a storage medium which is prone to risk of getting lost or stolen, and when it is to be transmitted through a network using mechanism which is not much secure. In [16], users data need to pass through two servers which is a major drawback as it can be retrieved by some attacker during transmission. Proposed scheme rectifies this problem by encrypting data at clients end such that data will be transferred over the network only after getting encrypted which enhances its security. The model in [10] stores complete data as a single piece at the server. If the server security is breached, data is vulnerable. Proposed scheme allows the user to divide the data in parts and store it at different cloud servers of his choice. In this way the user have the knowledge about the location of his data and if the security of any server is breached, data is still secure. If such an incident occurs and some modifications are to the file, an integrity verification using SHA-2 informs the user about it.

VIII. Conclusion and Future Work

The proposed scheme solves the problem of data security in a distributed storage system and file division takes the security to next level. User has privilege to store his data at the server of his choice so that he'll be able to track the server where a security breach occurs. It provides a base for enhancing the security of confidential data and future enhancements can be made to it. A combination of various encryption algorithms can be used which will make it almost impossible for an unauthorized person to access the confidential data. The model can further be improved by incorporating schemes like PDP, ORUTA etc. to solve issues like bandwidth, cost etc.

References

[1] *The NIST Definition of Cloud Computing*. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

[2] H. Qusay, "Demystifying cloud computing," *The Journal of Defense Software Engineering (CrossTalk)*, 2011.

[3] *Cloud Computing Privacy Concerns on Our Doorstep*. [Online]. Available: http://cacm.acm.org/magazines/2011/1/103200-cloud-computing-privacy-concerns-on-our-doorstep/fulltext

[4] A. Chhibber and S. Batra, "Security analysis of cloud computing," *International Journal of Advanced Research in Engineering and Applied Sciences*.

[5] W. T. Liu, "Research on cloud computing security problem and strategy," in *Proc. 2nd International Conference on Consumer Electronics, Communications and Networks*, 2012, pp. 1216-1219.

[6] B. Thiyagarajan and R. Kamalakannan, "Data integrity and security in cloud environment using aes algorithm," in *Proc. International Conference on Information Communication and Embedded Systems*, 2014, pp. 1-5.

[7] F. A.-J. Mohammed and A. Zainal, "Data integrity and privacy model in cloud computing," in *Proc. International Symposium on Biometrics and Security Technologies*, 2014, pp. 280-284.

[8] P. Ora and P. R. Pal, "Data security and integrity in cloud computing based On RSA partial homomorphic and MD5 cryptography," in *Proc. International Conference on Computer, Communication and Control*, 2015, pp. 1-6.

[9] G. Raj, R. C. Kesireddi, and S. Gupta, "Enhancement of Security Mechanism for Confidential Data using AES-128, 192 and 256bit Encryption in Cloud," Next Generation Computing Technologies (NGCT), 2015 1st International Conference, pp. 374-378.

[10] N. Surv, B. Wanve, R. Kamble, S. Patil, and J. Katti, "Framework for client side AES encryption technique in cloud computing," in *Proc.*

IEEE International Advance Computing Conference (IACC), 2015, pp. 525-528.
[11] V. S. Mahalle and A. K Shahade, "Enhancing the data security in cloud by implementing hybrid (RSA & AES) encryption algorithm," in *Proc. 2014 International Conference on Power, Automation and Communication*, pp. 146-149.
[12] R. Kaur and R. P. Singh, "Enhanced cloud computing security and integrity verification via novel encryption techniques," in *Proc. International Conference on Advances in Computing, Communications and Informatics*, 2014, pp. 1227-1233.
[13] L. Arockiam and S. Monikandan, "Efficient cloud storage confidentiality to ensure data security," in *Proc. International Conference on Computer Communication and Informatics*, 2014, pp. 1-5
[14] M. S. G. Prasad, H. R. Nagesh, and L. Dharmanna, "Ensuring data storage in cloud computing for distributed using high security password," in *Proc. National Conference on Research & Technology in the Coming Decades*, 2013, pp. 1-4.
[15] K. Chachapara and S. Bhadlawala, "Secure sharing with cryptography in cloud computing," in *Proc.* 2013 Nirma University International Conference on Engineering (NUiCONE), pp. 1-3.
[16] P. Rewagad and Y. Pawar, "Use of digital signature with diffie hellman key exchange and AES encryption algorithm to enhance data security in cloud computing," in *Proc.* 2013 *International Conference on Communication Systems and Network Technologies*, pp. 437-439.
[17] M. Z. Meetei and A. Goel, "Security issues in cloud computing," in *Proc. 2012 5th International Conference on BioMedical Engineering and Informatics*.
[18] A. Kumar, B. G. Lee, H. J. Lee, and A. Kumari, "Secure storage and access of data in cloud computing," in *Proc.* 2*012 International Conference on ICT Convergence*, pp. 336-339.
[19] M. Hamdi, "Security of cloud computing, storage, and networking," in *Proc. 2012 International Conference on Collaboration Technologies and Systems*, pp. 1-5.
[20] E. M. Mohamed, H. S. Abdelkader, and S. EI-Etriby, "Enhanced data security model for cloud computing," in *Proc.* 2012 *8th International Conference on Informatics and Systems (INFOS)*, pp. CC12-CC17.
[21] J. Grover, Shikha, and M. Sharma, "Cloud computing and its security issues – A review," in *Proc. 2014 International Conference on Computing, Communication and Networking Technologies*, pp. 1-5.
[22] W. Stalling, *Cryptography and Network Security: Principles and Practice*, 5th ed., 2011.
[23] Cryptography: What are the advantages and disadvantages of AES over Triple-DES? [Online]. Available: https://www.quora.com/Cryptography-What-are-the-advantages-and-disadvantages-of-AES-over-Triple-DES
[24] Description of SHA-1 and SHA-256. [Online]. Available: http://www.quadibloc.com/crypto/mi060501.htm
[25] R. Masram, V. Shahare, J. Abraham, and R. Moona, "Analysis and comparison of symmetric key cryptographic algorithms based on various file features," *International Journal of Network Security & Its Applications*, vol. 6, no. 4, July 2014.

Shubham Singh received the B.Tech degree in computer science and engineering from Invertis University, Bareilly, India, in 2015 and the M.Tech degree in computer science and engineering from the National institute of Technology Meghalaya, Shillong, India, in 2017.

He joined the Department of Computer Science and Engineering, National Institute of Technology Meghalaya, Shillong, India as a junior research fellow, in 2017. He has authored a paper in a peer-reviewed conference. His current research interests include cloud computing and hybrid automata.

Akhilendra Pratap Singh was born in India. He has completed his Ph.D in Service Oriented Architectures for wireless sensor networks from Indian Institute of Information Technology, Allahabad.

Currently Dr. Singh is serving in the Department of Computer Science and Engineering at National Institute of Technology, Meghalaya as an assistant professor. His research interests include service oriented architectures, wireless sensor networks and machine learning.