ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 3, March 2017

Secured Data Exchange in Cloud Computing

Aishwarya Chakraborty[1], Shreeya Kolhale[2], Divya Poddar [3], Reshma[4], Prof. K. B. Naik[5]

BE Students, Department of Information Technology, Bharati Vidyapeeth College of Engineering for Women, Savitribai Phule Pune University Pune, India1,[2,3,4]

Professor, Department of Information Technology, Bharati Vidyapeeth College of Engineering for Women, Savitribai Phule Pune University Pune, India[5]

ABSTRACT: Cloud computing is very popular nowadays. It provides storage at very low cost. It introduces few challenges related to data security[4] like data integrity, confidentiality and access control, along with that the keys generated for encrypting the data are also suffering of storage overheads as they are generated in large numbers. So, to address these issues we have proposed the scheme that uses Threshold Cryptography technique in which data owner divides users in groups and gives single key to each user group for decryption of data and, each user in the group shares parts of the key In this paper, the communication between three entities takes place The Data Owner (DO), Cloud Service Provider (CSP) and User. The main aim is to provide security and reduce key generation overheads.[1]

KEYWORDS: Data Owner (DO),Cloud Service Provider(CSP),User, Malicious user, access control, authentication, capability list, threshold cryptography.

1. INTRODUCTION

Cloud computing is a new and fast growing technology in field of computation and storage of data. It provides storage and computing as a service at very attractive cost. In our system, using cloud computing communication will be generated between Data Owner (DO), Cloud Service Provider (CSP) and the User[3]. The data is encrypted before it is uploaded on cloud. The users are divided into groups and each user receives the partial key. To check the integrity of data, message digest of each file is calculated before sending and after receiving data. There will be list called capability list which will be maintained to specify the access control rights of the users. Few algorithm will be used for the process of encryption and decryption of data and to generate one time shared session key between CSP and User to protect the data[3][4].

Many schemes are already in use to ensure these security requirements but they are suffering from collusion attacks of malicious users. Many algorithms are used to overcome these attacks but at times they all fail. So in this paper, we will be using the best provided algorithms yet, which will provide us the best result and cannot be attacked by malicious users. Now, let us discuss about the algorithm that are going to be used in this paper. There are few symmetric and asymmetric algorithms namely, MD5, RSA, Diffie-Hellman and Blowfish Algorithm as discussed in detail further.[3][13]

II. EXISTING SYSTEMS

In existing system cloud computing mostly two important security requirements are taken into consideration, one is to keep data secure and another is efficient access control for data. Some time we focus on data security we do not concentrate on system performance access control and have to focus on DO, CSP, users. Sometimes we use a large number of keys for encrypting the data. All keys are confidential so need to be kept secure and maintaining these keys are an additional work. These additional works affect the performance of the system. Hence it is needed to reduce

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 3, March 2017

these keys and our proposed system helps to reduce keys and provide data security as well as increases the performance of the system. Many schemes are suggested to meet these requirements.[2]

III. PROPOSED SYSTEM

1. We present a new approach which provides security for data outsourced at CSP. Some approaches are given to secure outsourced data but they are suffering from having large number of keys and collusion attack [2].

2. By employing the threshold cryptography at the user side, we protect outsourced data from collusion attack. Since, DO stores its data at CSP in encrypted form and, keys are known only to DO and respected users group, data confidentiality is ensured[1][2].

3. To ensure fine-grained access control of outsourced data, the scheme has used capability list[6]. Public key cryptography and MD5 ensure the entity authentication and data integrity respectively.

4. Public key cryptography and D-H exchange protected the data from outsiders in our approach. No of keys (because in threshold cryptography, there is a single key corresponding to each group) have reduced in the proposed scheme[2][5].

IV. ARCHITECTURE

As discussed earlier, we suppose that our model is composed of three entities: Data owner(DO), Cloud service provider(CSP) and many user associated with DO. So initially, all the users are registered at DO. We assume that users information/credentials are securely sent to DO. DO the divides the users in groups based on the threshold value which depends upon the space of cloud used and provides the encryption key for secure communication to user groups In response of registration. A user can only get data from CSP in confidential manner after successful authentication at CSP. DO has storage capacity to store some files and data and he can execute programs also at CSP to manage his files and data. We are using Diffie-hellman algorithm and public key cryptography to secure communication between CSP and User. The DO computes the digest of data to ensure the integrity of data using MD5 algorithm[5]. It encapsulates the digest and data using symmetric key algorithm. It provides strong integrity and confidentiality. DO fill the entries such as UID, FID and other details according to the type of project in capability list. DO then encrypt data with its private key and public key of CSP and sends it to CSP. This encryption ensures confidentiality and authentication between DO and CSP. When the CSP receives the encrypted data and capability list from DO it decrypt the data using its own private key and public key of DO and stores it in its storage. CSP then updates the encrypted data and capability list[7][8]. As the data is encrypted using symmetric key which is known only to the DO and respected user group, CSP can't send see the data. The CSP sends the encrypted symmetric key to respective user group after updating. Users of user group then decrypt the data and get their own parts of symmetric key. After getting the details user can request CSP for data. After receiving the data the user decrypts the message and calculates the digest of data to check the integrity and whether it matches with the digest calculated before sending the data[9][11]. If it matches, the file received is original otherwise the file is modified by outsider. The user sends error notification message to the DO of the file is found to be modified. Let us understand the total process by the diagram given below[1].

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 3, March 2017

V. ARCHITECTURE DIAGRAM

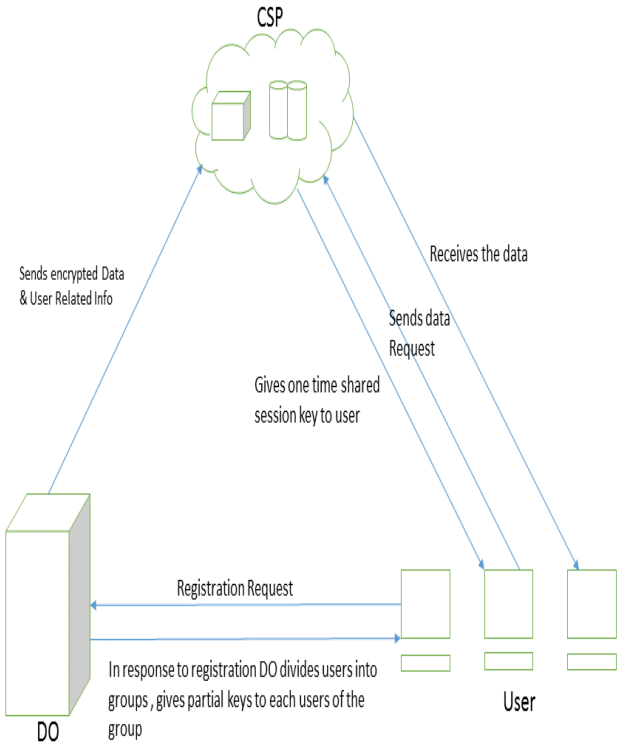


Fig No 01 System Architecture

VI. ALGORITHMS USED

In this paper, we are going to use few symmetric and asymmetric algorithm. The algorithms that we are going to use are Diffie-hellman, RSA, Blowfish and MD5. It is explained in detail below.

I. Blowfish Algorithm

Blowfish is an another algorithm designed by Bruce Schneier in 1993 to replace DES. This symmetric cipher splits messages into blocks of 64 bits and encrypts them individually. Blowfish can be found in software categories ranging from e-commerce platforms for securing payments to password management tools, where it used to protect passwords. It's definitely one of the more flexible encryption methods available.[12]

It will convert a key of at most 448 bits into several subkey arrays totalling 4168 bytes. These keys are generated earlier for any data encryption or decryption.

The p-array consists of 18, 32-bit subkeys:

P1,P2,……..,P18

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 3, March 2017

Four 32-bit S-Boxes consists of 256 entries each:

S1,0, S1,1,……. S1,255

S2,0, S2,1,…….. S2,255

S3,0, S3,1,…….. S3,255

S4,0, S4,1,.....S4,255

The subkeys are calculated using the Blowfish algorithm as follows:

1. Initialize first the P-array and then the four S-boxes, in order, with a fixed string. This string consists of the hexadecimal digits of pi (less the initial 3): P1 = 0x243f6a88, P2 = 0x85a308d3, P3 = 0x13198a2e, P4 = 0x03707344, etc.

2. XOR P1 with the first 32 bits of the key, XOR P2 with the second 32-bits of the key, and so on for all bits of the key (possibly up to P14). Repeatedly cycle through the key bits until the entire P-array has been XORed with key bits. (For every short key, there is at least one equivalent longer key; for example, if A is a 64-bit key, then AA, AAA, etc., are equivalent keys.)

3. Encrypt the all-zero string with the Blowfish algorithm, using the subkeys described in steps (1) and (2).

4. Replace P1 and P2 with the output of step (3).

5. Encrypt the output of step (3) using the Blowfish algorithm with the modified subkeys.

6. Replace P3 and P4 with the output of step (5).

II. MD-5 Algorithm

The MD5 function is a cryptographic algorithm that takes an input of arbitrary length and produces a message digest that is 128 bits long. The digest is sometimes also called the "hash" or "fingerprint" of the input. MD5[12] is used in many situations where a potentially long message needs to be processed and/or compared quickly. The most common application is the creation and verification of digital signatures. MD5 was designed by well-known cryptographer Ronald Rivest in 1991. In 2004, some serious flaws were found in MD5. The complete implications of these flaws has yet to be determined.

Preparing the input

The MD5 algorithm first divides the input in blocks of 512 bits each. 64 Bits are inserted at the end of the last block. These 64 bits are used to record the length of the original input. If the last block is less than 512 bits, some extra bits are 'padded' to the end.

Next, each block is divided into 16 words of 32 bits each. These are denoted as M_0 ... M_{15}.

MD5 helper functions:-

a. The buffer

MD5 uses a buffer that is made up of four **words** that are each 32 bits long. These words are called A, B, C and D. They are initialized as

word A: 01 23 45 67
word B: 89 ab cd ef
wor C: fe dc ba 98
word D: 76 54 32 10

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 3, March 2017

b. The table

MD5 further uses a table K that has 64 elements. Element number i is indicated as K_i. The table is computed beforehand to speed up the computations. The elements are computed using the mathematical sin function:
$K_i = abs(sin(i + 1)) * 2^{32}$

c. Four auxiliary functions

In addition MD5 uses four auxiliary functions that each take as input three 32-bit words and produce as output one 32-bit word. They apply the logical operators and, or, not and xor to the input bits.

F(X,Y,Z) = (X and Y) or (not(X) and Z)
G(X,Y,Z) = (X and Z) or (Y and not(Z))
H(X,Y,Z) = X xor Y xor Z
I(X,Y,Z) = Y xor (X or not(Z))

III. RSA algorithm

RSA involves a public key and private key. The public key can be known to everyone, it is used to encrypt messages. Messages encrypted using the public key can only be decrypted with the private key[12]. The keys for the RSA algorithm are generated the following way:

1. Select two prime numbers a and b where a is not equal to b.Choose t
2. Calculate n= a*b
3. Calculate the totient: .as (a-1)*(b-1).

Select e such that, e is relatively prime to the totient calculated above ie gcd of e and the totient is 1.

4. Calculate of such that $d= e^{-1}$ mod totient =1.
5. Private key ={d,n},public key ={e,n}.
6. Find out cipher text using the formula:-

$C=P^e$ mod n,where P<n and
C=Ciphertext,P=Plaintext,e=Encryption Key and n=Block size.

7. $P=C^d$ mod n.,d=decryption key.

IV. D-H Algorithm

Diffie-Helman is a way of generating a shared secret between two people in such a way that the secret can't be seen by observing the communication.
The basic idea works works like this:

1. I come up with two prime numbers **g** and **p** and tell you what they are.
2. You then pick a secret number (**a**), but you don't tell anyone. Instead you compute $\mathbf{g}^{\mathbf{a}}$ mod **p** and send that result back to me. (We'll call that **A** since it came from **a**).
3. I do the same thing, but we'll call my secret number **b** and the computed number **B**. So I compute $\mathbf{g}^{\mathbf{b}}$ mod **p** and send you the result (called "**B**")
4. Now, you take the number I sent you and do the exact same operation with it. So that's $\mathbf{B}^{\mathbf{a}}$ mod **p**.
5. I do the same operation with the result you sent me, so: $\mathbf{A}^{\mathbf{b}}$ mod **p**.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

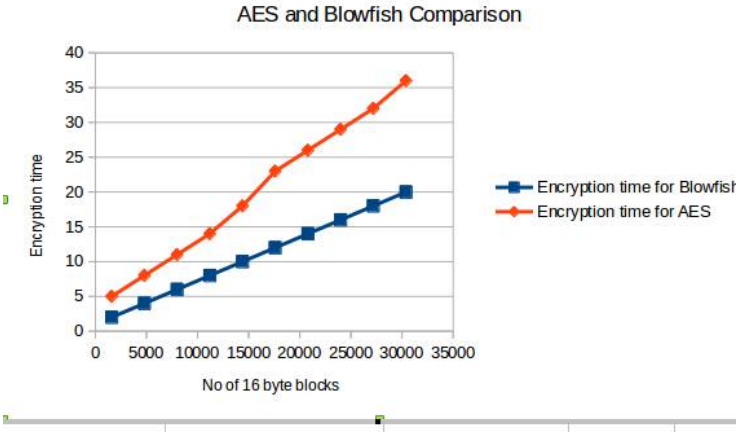
Website: www.ijirset.com

Vol. 6, Issue 3, March 2017

VII. EXPERIMENTAL EVALUATION

a. AES and Blowfish comparison

No of 16 byte blocks	Encryption time for Blowfish	Encryption time for AES
1600	2	5
4800	4	8
8000	6	11
11200	8	14
14400	10	18
17600	12	23
20800	14	26
24000	16	29
27200	18	32
30400	20	36



It is observed that Blowfish has better perormance time as compared to AES ,also blowfish is more secure. Therfore,we have used blowfish as our symmetric algorithm instead of AES.The other algorithms used are MD5 ,RSA and Diffie Hellman. We have seen that number of keys (because in threshold cryptography, there is a single key corresponding to each group) have reduced in the proposed scheme. Hence, reduces the maintenance and security concerns of keys and reduces the additional computation time.

VIII. CONCLUSION

In this paper, we presented a new approach which provides security for data outsourced at CSP. Some approaches are given to secure outsourced data but they are suffering from having large number of keys and collusion attack. By employing the threshold cryptography at the user side, we protect outsourced data from collusion attack. Since, DO stores its data at CSP in encrypted form and, keys are known only to DO and respected users group, data confidentiality is ensured. To ensure fine-grained access control of outsourced data, the scheme has used capability list. Public key cryptography and MD5 ensure the entity authentication and data integrity respectively. Public key cryptography and D-H exchange protected the data from outsiders in our approach. No of keys (because in threshold cryptography, there is a single key corresponding to each group) have reduced in the proposed scheme.

ACKNOWLEDGEMENTS

I would like to thank Dr. Prof. Mrs. D. A. Godse, Head of the Department of Information technology engineering, for her valuable suggestions in completing this work successfully. We would like to thank other faculty members for their support and encouragement.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 3, March 2017

REFERENCES

[1]Sushil Kr Saroj, Sanjeev Kr Chauhan, Aravendra Kr Sharma Sundaram Vats,"Threshold Cryptography based secured data exchange in cloud computing,2015IEEE International conference on computational intelligence and communication technology.
[2] J. Do, Y. Song, and N. Park, "Attribute Based Proxy Re-encryption for Data Confidentiality in Cloud Computing Environments," Computers, Networks, Systems and Industrial Engineering (CNSI), 2011 First ACIS/JNU International Conference on, vol., no., pp.248-251, 23-25 May 2011.
[3] N. Bennani, E. Damiani, and S. Cimato, "Toward Cloud-Based Key Management for Outsourced Databases," Computer Software and Applications Conference Workshops (COMPSACW), 2010 IEEE 34th Annual, vol., no., pp.232-236, 19-23 July 2010.
[4] S. Sanka, C. Hota, and M. Rajarajan, "Secure data access in cloud computing," Internet Multimedia Services Architecture and application (IMSAA), 2010 IEEE 4th International Conference on, vol., no., pp.1-6, 15-17 Dec. 2010.
[5] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," in Proc. of NDSS'05, 2005.
[6] C. Hota, S. Sanka, M. Rajarajan, and S. Nair, "Capability-Based Cryptographic Data Access Control in Cloud Computing," Int. J. Advanced Networking and Applications Volume: 01 Issue: 01 Page: (2011).
[7] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," Association for Computing Machinery, in Proc. of CCS'06, 2006.
[8] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing," in Proc. of IEEE INFOCOM 2010, 2010.
[9] S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Over-encryption: Management of access control evolution on outsourced data," in Proc. of VLDB'07, 2007.207
[10] R. S. Fabry, "Capability-Based Addressing," in Communications of the ACM, 17(7), July 1974, pp. 403-412.
[11] A. Shamir, "How to share a secret," Communications of the ACM, v.22 n.11, p.612-613, Nov. 1979. [Online]. Available:http://portal.acm.org/citation.cfm?id=359168.359176.
[12]"Information and Cyber Security" Santosh A Dharad,Sahani Sarangi