ISSN 2395-1621

Threshold Cryptography Based Data Security in Cloud Computing

[#1]Mr Y .K. Sharma, [#2]Rohit Singh, [#3]Ujjwal Priyadarshi,
[#4]Yogesh N. Rathod

[3]priyadarshiujjwal@gmail.com

[#1]Asst Professor, Department of Computer
[#234]Student, Department of Computer

VIIT, Pune.

ABSTRACT

Cloud computing is very popular in organizations and institutions because it provides storage and computaion services at a very low price. But it has some challenges about ensuring the data confidentiality, data integrity and data access control. People have presented some approaches but they lacked in some ways for example data confidentiality violation because of the collusion attack and heavy computation which are caused by the large number of keys. To resolve these problems we propose a technique which makes use of the threshold cryptography in which data owner will divide the users in groups and will give a single key to each user group for the decryption of data and, each user in the group will share a part of the key. This scheme provides the strong data confidentiality as well as reduces the number of keys.

Keywords: Outsourced data, malicious outsiders, access control, authentication, threshold cryptography.

ARTICLE INFO

Article History

Received: 18th May 2017

Received in revised form :

18th May 2017

Accepted: 21st May 2017

Published online :

25th May 2017

I. INTRODUCTION

Cloud computing is a popular technology in the field of data computation. It provides storage and computing as a service at a very reasonable price. It delivers services according to three basic service models: infrastructure as a service (IaaS), platform as a service (PaaS) and software as a service (SaaS). Storage as a service is basically a platform as a service. Cloud computing gives service on demand, self-service, It is independent of the location, It has rapid elasticity and a measured scale service. This makes cloud significant. Industries and institutions are using these characteristics of cloud computing and increasing their profit.

Data security is a major concern in the way of cloud computing. People are still afraid of using the cloud computing and some think of cloud as not a safe place and once you upload your data on cloud, you do not have at all
Control over it [8][9]. They are more or less right. Data of data owners are processed and stored at external servers. So, confidentiality, integrity and access of data become more vulnerable.

The external servers are under total control of commercial service providers that is the reason why data owner can't trust on them as they are able to access and use the data for their own benefits and can spoil businesses of data owner [4]. Data owner can't keep belief on users because they could be malicious. So the Data Owner needs to be assured of the thing that the data it is going to share must remain secured and not accessed by the unauthorized people. Confidentiality of the data may get violated through the collusion attack of the malicious users and service providers.

In this approach, there are basically three components: Data Owner(DO), Cloud Service Provider (CSP) and Users. Users are divided in the groups on some basis such as location, department and, corresponding to each group, there is a single key which will be used to encrypt and decrypt the data. Each user in the group will get a part of the key. Data will be decrypted only when at least threshold number of users will be available. This approach provides data confidentiality as well as reduces the number of keys. The approach has used the modified Diffie-Hellman algorithm to generate one time shared session-key between Cloud Service Provider and user for the data security from outsiders.

II. RELATED WORK

Data confidentiality and access control are two fundamental security requirements for outgoing data in cloud computing. When we care of more for the security of data, we might neglect the performance of the systems (Data Owner, CSP, users). For example, to keep the data more secured, we sometimes use a large number of keys. Keys are confidential, so we need to secure and maintain these keys which are additional overhead which reduces the system performance. So, it is required to reduce the number of keys. This creates a need of a scheme that is able to provide data security as well as maintain the performance.

Symbol	Description
DO	Data Owner
CSP	Cloud Service Provider
AR	Access Rights
CPList	Capability List
UID	User ID
FID	File ID

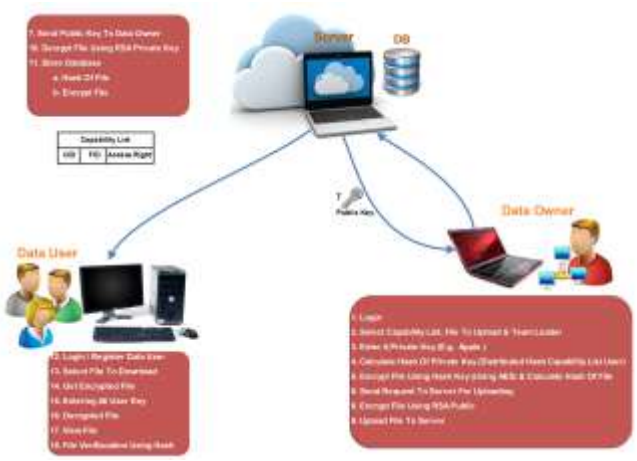
The scheme proposed in [13] is the group-key scheme. In group-key scheme, there is a single key corresponding to each group of users for the data decryption and all users of the group have the knowledge of the key. Thus number of keys is reduced but there is an issue of collusion attack of CSP and a user because a single malicious user can leak the whole data of the group to CSP. Data Owner cannot trust CSP since it can use the data owner's data for its commercial benefits.

The scheme proposed in [4] tried to achieve data confidentiality and access control. In this scheme, data encryption is done by using the symmetric keys and symmetric keys are known only to the data owner and corresponding data users. The data after encryption process are stored at CSP. CSP is totally unable to see the data stored at it as data are encrypted. Data are further encrypted by one time secrete session-key shared between CSP and user by the modified Diffie-Hellman protocol for the purpose of data protection from the outsiders during the data transmission process between the CSP and user. This scheme no doubt provides whole data security but there is associated a key corresponding to each user and users may be large in number in some applications. So, number of keys may increase. This increases the maintenance and security concerns of keys.

Communication model of the proposed scheme somehow matches with it [4] but proposed scheme is more secure and reduces number of keys. The proposed scheme is useful for those applications where works are usually done in the team and group such as in the software companies. It is applicable all where users can be grouped on some basis and can apply threshold cryptography technique. Such as software and hardware industries, institutes, banks and medicals fields.

III. MODEL AND ASSUMPTIONS

Our model consists of three modules: a CSP, a Data Owner (DO) and many users associated with Data Owner (DO). Initially, all users will get registered at Data Owner using the registration process during the registration process users will send their details to the Data Owner. Data Owner (DO) will divide users into the groups and provide encryption keys, tokens, algorithms for the secure communication to user groups in response of registration. A user will get the data from CSP in a confidential manner after getting successfully authenticated at CSP. We assume that no one can break the security of CSP. We also assume that the algorithm which is used to generate the secret keys for encrypting the data is secure at Data Owner. Data Owner has the storage capacity for storing some files and data and, he can execute programs also at CSP to manage his files and data. We use the modified Diffie-Hellman algorithm and public cryptography to secure communication between CSP and user. Modified Diffie-Hellman protocol is used to create one time session-key between CSP and user. Fig.1 illustrates the secure communication between entities in the proposed schema.



IV. PROPOSED SCHEME

We propose a complete model for secure communication and secure data access. There are three algorithms in the proposed scheme. Algorithm 1 describes secure communication of data between Data Owner and CSP moreover this algorithm insures data confidentiality and, authentication of Data Owner and CSP. Algorithm 2 describes procedures which Data Owner and CSP apply after a new file creation in respect. Algorithm 3 describes secure communication of data between CSP and user. In this algorithm user's authorization is also checked. This algorithm also describes the threshold cryptography method for decryption of a user's file and it is applied at the user side where number of keys is reduced which ensures of no threat of collusion attack as in group-key scheme.

For the better understanding of proposed scheme let's take an example of real life scenario, Data Owner may be a software industry who stores its data on to the CSP and the users may be its employees who view their data from the CSP. Data Owner divides users in groups on some basis such as project basis and encrypts the data of each group with a single symmetric key and, it gives parts of the symmetric key to each user of the group. Data Owner then fills the entries such as User ID, File ID and Access Rights in Capability List corresponding to each new user. Data Owner then encrypts Capability List and encapsulated things with its private key after that public key of CSP and, then sends all things to CSP. These encryptions ensure confidentiality and authentication between Data Owner and CSP.

Algorithm 1:

Step 1: Data owner selects the file to upload.
Data owner encrypts it with AES.
SHA-1 will calculate hash of the AES key; it will break the key into a number of users.

Step 2: CSP will store the file which is encrypted and Capability List which are received from Data owner.

Step 3: CSP will update the Encrypted File List and capability List

Algorithm 1 describes the process what CSP does after getting the encrypted file and the Capability List from the Data Owner (DO). The file is encrypted using AES algorithm and to make it more secured, AES key is encrypted and divided into some parts using SHA-1.The file is again encrypted with RSA public algorithm. CSP decrypts the message using its own private key and the public key of Data Owner (DO) and stores the encrypted file and Capability List in its storage. CSP will update the list of the Encrypted Files and Capability List. Since file is encrypted using the AES key and only the Data Owner and respected user group have the knowledge of it, CSP is not able to see the data even though user's credential comes through it.

Algorithm 2:

Step 1: Data Owner updates Capability List.

Step 2: Now, Data Owner encrypts the Capability List, Encrypted File, AES key and sends these to the CSP.

Step 3: CSP Updates its copy of the Capability List, Encrypted File List and sends symmetric key to the indented user group.

Step 4: Now, the user can send actual access request for that file directly to CSP.

Algorithm 2 describes the procedure needed after a new File creation. When a new File is created, Data owner fills entries for that File in Capability List containing User ID, File ID and Access Rights. Data Owner generates a symmetric key and encrypts File with that symmetric key. Now, Data Owner encrypts the updated Capability List, Encrypted File and symmetric key with its private key and sends these to the CSP. When CSP receives these, it updates Capability List, Encrypted File List and sends encrypted symmetric key to respective user group. Users of the user group then decrypt the message using their own parts of the symmetric key.

Algorithm 3 describes how data exchange takes place in a secure manner between CSP and the user by use of modified Diffie-Hellman algorithm. We called it modified D-H algorithm as we encrypt the D-H parameters using the public key of one side and, using nonce in each direction during session key generation and data transfer. It helps to counter the man-in-the middle attack. After available of keys and tokens, the user may request for data to CSP. CSP initiates modified D-H key exchange with the user, if request is authentic. We assume that the session key is shared between CSP and the user by modified Diffie-Hellman algorithm. Now, CSP encrypts the encrypted File using RSA algorithm with the shared session key and sends it to the user. This over encryption ensures the confidentiality of the message between cloud service provider and the user. The user then decrypts the message.

Algorithm 3: Algorithm for secure data exchange between CSP and User by using Modified D-H key exchange

Step 1: User sends data access request to CSP

Step 2: CSP matches User ID, File ID, Access Rights with Capability List stored at it.

Step 3: CSP initiates D-H exchange with that User and shares one time shared session key.

Step 4: CSP encrypts the encrypted File Using RSA algorithm and a part of the decryption key is sent to each user to their e mail id.

Step 5: Number of users get their individual parts of the decryption key on their e mail provided, when all the keys are entered for the respective users, file gets decrypted and downloaded.

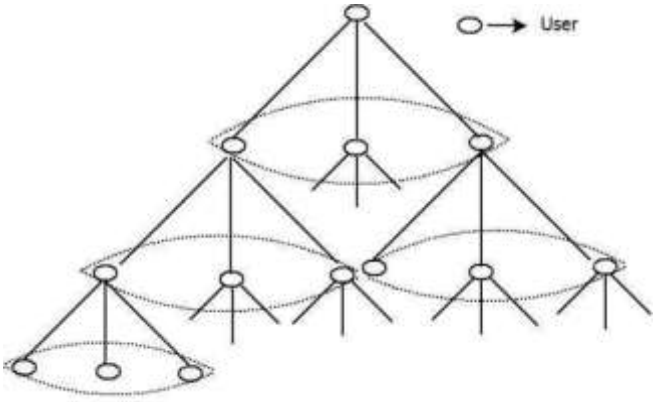
Thus, the threshold cryptography based data security can be provided to the users, institutions, companies, any other

field or sector where there is need for the data security as well as data confidentiality and data integrity using these algorithms that describe the step by step procedure for the threshold cryptography and accessing the data. It is called threshold cryptography because the key for the purpose of decryption of the important data is sent to the threshold number of users.

Until and unless the threshold number of users enters their parts of the key sent to their E-mail id provided, the data files will not get decrypted and thus not downloaded. Thus the data files are safe and confidential and remain into the encrypted form until all the threshold number of users key in their parts of the decryption key. After they enter their respective parts of the decryption key the data files will get decrypted as well as downloaded to the machine. Now the users can have the access to the data files.

V. HIERARCHY MANAGEMENT OF ACCESS RIGHTS

In this section, we describe hierarchy of access rights of users. Following figure gives an idea of the description of it. Here, users which are encircled in the same group are having the same access rights. In other way, we can say that they are sharing key components of the key with which data of that group are encrypted. Along with this, each user can also see data of all the groups which are formed with users beneath him in the hierarchy. Here, we are calling child group for such users group but the user uses dummy key for it and this key is not shared with any others. When a user wants to see data of his child group, he encrypts the encrypted data of child group with his dummy key and sends encrypted data to that child group for decryption. The user then decrypts it with his dummy key and gets data.



Hierarchy of Access Rights

VI. SECURITY AND PERFORMANCE ANALYSIS

A. Security Analysis

1) Data confidentiality: In this scheme, Data Owner stores its data at Cloud Service Provider in an encrypted form. Since, data are encrypted by the symmetric keys which available with only Data Owner and respective users group, CSP is totally unable to see the actual data. After validation of user request, Cloud Service Provider sends the encrypted data to the user. To protect the data from outsiders CSP again encrypts the encrypted data by the onetime session key shared between CSP and user. In the proposed scheme, no member of any group knows about the whole key (due to threshold cryptography). They have the knowledge of only a part of a key. Thus collusion attack of CSP and users is impossible.

2) Entity Authentication: In the proposed scheme, user is authenticated at Data Owner when he sends his personal details to Data Owner by encrypting its own private key during registration. DO is authenticated at CSP when it sends capability list and encrypted message digest and data to CSP by encrypting its own private key. User is authenticated at CSP when user's ID and password match with user's ID and password stored at the database of CSP.

3) Data Access Control: To ensure data access control, the proposed scheme uses capability list in the approach. Capability list basically contains the User ID, File ID and Access Rights. Only Data Owner has rights to perform any operation on it. CSP only can read it for the purpose of secure data access. CSP sends only those data to users what are in their access rights. In other way, users can access those data which are in their capability [15].

B. Performance Analysis

We have seen that Data Owner transferred maximum of its load and computation to CSP and did only necessary things by itself. No 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.

VII. CONCLUSION

In this paper, we presented a new approach which provides security for data outgoing at CSP. Some approaches are given to secure outsourced data but they have a large number of keys and collusion attack. By using the threshold cryptography, we can protect the outgoing data from collusion attack. Since, Data Owner stores its data at Cloud Service Provider in an encrypted form and, keys are available with only Data Owner and the respected users group, data confidentiality is ensured. To ensure access control of outgoing data, the approach has used capability list. Public key cryptography and SHA-1 guarantees the authentication of entity and data integrity respectively. The data are protected from outsiders in our approach. Number of keys (because in threshold cryptography, there is a single key corresponding to each group) have reduced in the proposed scheme.

REFERENCES

[1] 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.

[2] 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.

[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[16] S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. encryption for fine-grained access control of encrypted data," Association for Computing Machinery, in Proc. of CCS’06, 2006.

[8] T. Mather, S. Kumaraswamy, and S. Latif, “Cloud Security and Privacy," O'Reilly Media, Sep. 2009.

[9] A. T. Velte, T. J. Velte, and R. Elsenpeter, “Cloud computing a practical approach," Tata McGraw-Hill Edition, 2010, ISBN-13:978-0-07-068351-8.

[10] W. Stallings, “Cryptography and network security," LPE Forth Edition, ISBN-978-81-7758-774-6.

[11] G. Miklau, and D. Suciu, “Controlling access to published data using cryptography," in Proc. of 29th VLDB, Germany, Sept 2003.

[12] 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.

[13] H. Zhong, and H. Zhen, “An Efficient Authenticated Group Key Agreement Protocol," Security Technology, 2007 41st Annual IEEE International Carnahan Conference on, vol., no., pp.250-254, 8-11 Oct. 2007.

[14] S. K. Harit, S. K. Saini, N. Tyagi, and K. K. Mishra, “RSA Threshold Signature Based Node Eviction in Vehicular Ad Hoc Network," Information Technology Journal, 2012, ISSN 1812-5638, in Asian Network for Scientific Information.

[15] R. S. Fabry, “Capability-Based Addressing," in Communications of the ACM, 17(7), July 1974, pp. 403-412. Samarati, “Over-encryption: Management of access control evolution on outsourced data,” in Proc. of VLDB’07, 2007.

