RESEARCH ARTICLE OPEN ACCESS

Dual Access Control for Cloud-Based Data Storage and Sharing

Mrs R Jhansi Rani MCA [1], K Obulesu [2]

[1] Asst. Professor, Department Computer Application

[2] Student, Department of Computer Application

[1], [2] Chadalawada Ramanamma Engineering College (Autonomous)

ABSTRACT:

Cloud-based data storage service has drawn increasing interests from both academic and industry in the recent years due to its efficient and low cost management. Since it provides services in an open network, it is urgent for service providers to make use of secure data storage and sharing mechanism to ensure data confidentiality and service user privacy. To protect sensitive data from being compromised, the most widely used method is encryption. However, simply encrypting data (e.g., via AES) cannot fully address the practical need of data management. Besides, an effective access control over download request also needs to be considered so that Economic Denial of Sustainability (EDoS) attacks cannot be launched to hinder users from enjoying service. In this paper, we consider the dual access control, in the context of cloud-based storage, in the sense that we design a control mechanism over both data access and download request without loss of security and efficiency. Two dual access control systems are designed in this paper, where each of them is for a distinct designed setting. The security and experimental analysis for the systems are also presented.

Keywords: - Cloud-based data storage service has drawn increasing interests from both academic and industry in the recent years due to its efficient and low cost management

I. INTRODUCTION

In the recent decades, cloud-based storage service has attracted considerable attention from both academia and industries. It may be widely used in many Internet-based commercial applications (e.g., Apple iCould) due to its long-list benefits including access flexibility and free of local data management. Increasing number of individuals and companies nowadays prefer to outsource their data to remote cloud in such a way that they may reduce the cost of upgrading their local data management facilities/devices. However, the worry of security breach over out sourced data may be one of the main obstacles hindering Internet users from widely using cloud-based storage service. In many practical applications, outsourced data may need to be further shared with others. For example, a Dropbox user Alice may share photos with her friends. Without using data encryption, prior to sharing the photos, Alice needs to generate a sharing link and further share the link with friends. Although guaranteeing some level of access control over unauthorized users (e.g., those are not Alice's friends), the sharing link may be visible within the Dropbox administration level (e.g., administrator could reach the link). Since the cloud (which is deployed in an open network) is not be fully trusted, it is generally recommended to encrypt the data prior to being uploaded to the cloud to ensure data security and privacy. One of the corresponding solutions is to directly employ an encryption technique(e.g., AES) on the outsourced data before uploading to cloud, so that only specified cloud user (with valid decryption key) can gain access to the data via valid decryption. To prevent shared photos being accessed by the "insiders" of the system, a straightforward way is to designate the group of authorized data users prior to encrypting the data. In some cases, nonetheless, Alice may have no idea about who the photo receivers/users are going to be. It is possible that Alice only has knowledge of attributes w.r.t. photo receivers. In this case, traditional public key encryption (e.g., Paillier Encryption), which requires the encrypt or to know who the data receiver is in advance, cannot be leveraged. Providing policy-based encryption mechanism over the outsourced photos is therefore desirable, so that Alice makes use of the mechanism to define access policy over the encrypted photos to guarantee only a group of authorized users is able to access the photos. In a cloud-based storage service, there exists a common attack that is well-known as resource-exhaustion attack. Since a (public) cloud may not have any control over download

request(namely, a service user may send unlimited numbers of download request to cloud server), a malicious service user may launch the denial-of-service (DoS)/distributed denial-of-service (DDoS) attacks to consume the resource of cloud storage service server so that the cloud service could not be able to respond honest users' service requests. As a result, in the "pay-as-you-go" model, economic aspects could be disrupted due to higher resource usage. The costs of cloud service users will rise dramatically as the attacks scale up. This has been known as Economic Denial of Sustainability (EDoS) attack [32], [33], which targets to the cloud adopter's economic resources. Apart from economic loss, unlimited download itself could open a window for network attackers to observe the encrypted download data that may lead to some potential information leakage (e.g., file size). Therefore, an effective control over download request for outsourced (encrypted) data is also needed. In this paper, we propose a new mechanism, dubbed dual access control, to tackle the above aforementioned two problems. To secure data in cloud-based storage service, attribute-based encryption (ABE) [9] is one of the promising candidates that enable the confidentiality of outsourced data as well as fine-grained control over the outsourced data. In particular, Ciphertext-Policy ABE (CP-ABE) [5] provides an effective way of data encryption such that access policies, defining the access privilege of potential data receivers, can be specified over encrypted data. Note that we consider the use of CP-ABE in our mechanism in this paper. Nevertheless, simply employing CP-ABE technique is not sufficient to design an elegant mechanism guaranteeing the control of both data access and download request. A straw man solution to the control of download request is to leverage dummy ciphertexts to verify data receiver's decryption rights. It, concretely, requires data owner, say Alice, to upload multiple "testing" ciphertexts along with the "real" encryption of data to cloud, where the "testing" ciphertexts are the encryptions of dummy messages under the same access policy as that of the "real" data. After receiving a download request from a user, say Bob, cloud asks Bob to randomly decrypt one of the "testing" ciphertexts. If a correct result/decryption is returned (i.e. indicating Bob is with valid decryption rights), Bob is authorized by Alice to access the "real" data, so that the cloud allows Bob to download the corresponding ciphertext. Nevertheless, several disadvantages of the above approach may be identified as follows. First of all, the data owner, Alice, is required to encrypt a number of dummy ciphertexts under the same policy as the "real" ciphertext. This may yield a considerable computational overhead for Alice, which may bring inconvenience in practice, for example, Alice just wants to upload one photo to iCloud from her cellphone, but needs to prepare more than one ciphertexts. Second, all ciphertexts, including dummy ones, are uploaded to cloud at the same time. This inevitably imposes extra cost on network bandwidth (as well as prolonging data uploading time), which may not be applicable to some service users who se cellular network is under pay-as-you-go plan or equipped with old generation of broadband cellular network technology (e.g., 3G). Third, a data receiver/user, Bob, has to additionally decrypt a random-chosen "testing" ciphertext from cloud, as a test of his valid download request. As a result, Bob has to "pay" double (decryption price) for accessing to the "real" data, which again may not be scalable in resource constrained setting. Therefore, this paper raises the following question: "Does there exist a cloud-based mechanism supporting dual access control (over both fine-grained data access and download request) without loss of security and efficiency?"

II. RELATED WORK

To apply fine-grained policy-based control over encrypted data, ABE [9], [29] has been introduced in the literature. Concretely, ABE has two main research branches: one is CP-ABE, and the other is KP-ABE which refers to as keypolicy ABE. This paper mainly deals with the former. In a CP-ABE, decryption key is associated with attribute set and ciphertext is embedded with access policy. This feature makes CP-ABE quite suitable for secure cloud data sharing Note this is so because KP-ABE requires decryption key to be associated with access policy which yields heavy storage cost for cloud user. Since the introduction of seminal CP-ABE [9], many works have been proposed to employ CP-ABE in various applications, e.g., accountable and traceable CP-ABE [22], [23], [24], [25], multi-authority [10],

[17], outsourced CP-ABE [15], [16], [21], and extendable variants [14]. Although being able to support fine-grained data access,
CP-ABE, acting as a single solution, is far from practical and effective to hold against EDoS attack [11] which is the case of DDoS in the cloud setting [11], [39]. Several countermeasures to the attack [12], [33] have been proposed in the literature. But Xue et al. [38] stated that the previous works could not fully defend the EDoS attack in the algorithmic (or protocol) level, and they further proposed a solution to secure cloud data sharing from the attack. However, [8] suffers from two disadvantages. First, the data owner is required to generate a set of challenge ciphertexts in order to resist the attack, which enhances its computational burden. Second, a data user is required to decrypt one of the challenge ciphertexts as a test, which costs a plenty of expensive operations (e.g., pairing). Here the computational complexity of both parties is inevitably increased and meanwhile, high network bandwidth is required for the delivery of ciphertexts. The considerable computational power of cloud is not fully considered in [38]. In this paper, we will present a new solution that requires less computation and communication cost to stand still in front of the EDoS attack. Recently, Antonis Michalas [20] proposed a data sharing protocol that combines symmetric searchable encryption and ABE, which allows users to directly search over encrypted data. To implement the functionality of key revocation in ABE, the protocol utilizes SGX to host a revocation authority. Bakas and Michalas [3] later extended the protocol in [20] and proposed a hybrid encryption scheme that reduces the problem of multi-user data sharing to that of a single-user. In particular, the symmetric key used for data encryption is stored in an SGX enclave, which is encrypted with an ABE scheme. Similar to [20], it deals with the revocation problem in the context of ABE by employing the SGX enclave. In this work, we employ SGX to enable the control of the download request (such that the DDoS/EDoS attacks can be prevented). In this sense, the purpose and the technique of ours are different from that of the protocols in [3], [20].

III. EXISTING SYSTEM

In a cloud-based storage service, there exists a common attack that is well-known as resource-exhaustion attack. Since a (public) cloud may not have any control over download request (namely,a service user may send unlimited numbers of download request to cloud server), a malicious service user may launch the denial-of-service (DoS)/distributed denial-of-service (DDoS) attacks to consume the resource of cloud storage service server so that the cloud service could not be able to respond honest users' service requests. As a result, in the "pay-as-you-go" model, economic aspects could be disrupted due to higher resource usage. The costs of cloud service users will rise dramatically as the attacks scale up. This has been known as Economic Denial of Sustainability (EDoS) attack, which targets to the cloud adopter's economic resources. Apart from economic loss, unlimited download itself could open a window for network attackers to observe the encrypted download data that may lead to some potential information leakage (e.g., file size). Therefore, an effective control over download request for outsourced (encrypted) data is also needed.

DISADVANTAGES:-

User deduplication on the client-side, cannot generate a new tag when they update the file. In this situation, the dynamic Ownerships would fail. As a summary, existing dynamic Ownerships cannot be extended to the multi-user environment. Whenever data is transformed, concerns arise about potential loss of data. By definition, data deduplication systems store data differently from how it was written. As a result, users are concerned with the integrity of their data. One method for deduplicating data relies on the use of cryptographic hash functions to identify duplicate segments of data. If two different pieces of information generate the same hash value, this is known as a collision. The probability of a collision depends upon the hash function used, and although the probabilities are small, they are always non zero.

IV. PROPOSED SYSTEM

In this paper, we propose a new mechanism, dubbed dual access control, to tackle the above aforementioned two problems. To secure data in

cloud-based storage service, attribute-based encryption (ABE) [9] is one of the promising candidates that enables the confidentiality of outsourced data as well as fine-grained control over the outsourced data. In particular, Ciphertext-Policy ABE (CP-ABE) [5] provides an effective way of data encryption such that access policies, defining the access privilege of potential data receivers, can be specified over encrypted data. Note that we consider the use of CP-ABE in our mechanism in this paper. Nevertheless, simply employing CP-ABE technique is not sufficient to design an elegant mechanism guaranteeing the control of both data access and download request.

ADVANTAGES:-

system has two advantages. Firstly, it can be used to confidentially share data with users by specifying access policies rather than sharing decryption keys. Secondly, it achieves the standard notion of semantic security for data confidentiality while existing systems only achieve it by defining a weaker security notion. In addition, we put forth a methodology to modify a ciphertext over one access policy into ciphertexts of the same plaintext but under other access policies without revealing the underlying plaintext.

V. MODULES IMPLEMENTATION

DATA OWNER:

In this module, initially the data owner has to register to the cloud server and get authorized. After the authorization from cloud data owner will encrypt and add file to the cloud server where in after the addition of file data owner requests the content key and the master secret key to the authority for the file he uploaded and finds Find deduplication ,only after the keys generated the file is uploaded to the cloud server. After the uploading of the file the data owner will have to provide download and the search permission for individual file for the users to perform search and download.

CLOUD SERVER

The cloud server manages a cloud to provide data storage service. Data owners encrypt their data files and store them in the cloud for sharing with cloud End users. To access the shared data files users will request the permission of content key and the MSK master secret key. And the cloud will provide the permission .and also views all the transactions and attackers related to the files.

AUTHORITY

Authority generates the content key and the secret key requested by the end user.

Authority can view all files with the content key and master secret key generated with the corresponding data owner details of the particular file.

END USER

User has to register and login for accessing the files in the cloud. User is authorized by the cloud to verify the registration. User has to request for the MSK master secret key and content key to download the file. User can only download and serach the file if the data owner of the particular file has provided the permissions.

VI. CONCLUSION

We addressed an interesting and long-lasting problem in cloud-based data sharing, and presented two dual access control systems. The proposed systems are resistant to DDoS/EDoS attacks. We state that the technique used to achieve the feature of control on download request is “transplantable” to other CPABE constructions. Our experimental results show that the proposed systems do not impose any significant computational and communication overhead (compared to its underlying CP-ABE building block).

REFERENCES

[1] Joseph A Akinyele, Christina Garman, Ian Miers, Matthew W Pagano, Michael Rushanan, Matthew Green, and Aviel D Rubin. Charm: a framework for rapidly prototyping cryptosystems. Journal of Cryptographic Engineering, 3(2):111–128, 2013.

[2] Ittai Anati, Shay Gueron, Simon Johnson, and Vincent Scarlata. Innovative technology for cpu

based attestation and sealing. In Workshop on hardware and architectural support for security and privacy (HASP), volume 13, page 7. ACM New York, NY, USA, 2013.

[3] Alexandros Bakas and Antonis Michalas. Modern family: A revocable hybrid encryption scheme based on attribute-based encryption, symmetric searchable encryption and SGX. In SecureComm 2019, pages 472–486, 2019.

[4] Amos Beimel. Secure schemes for secret sharing and key distribution. PhD thesis, PhD thesis, Israel Institute of Technology, Technion, Haifa, Israel, 1996.

[5] JohnBethencourt,AmitSahai,andBrentWaters. Ciphertext-policy attribute-based encryption. In S&P 2007, pages 321–334. IEEE, 2007.

[6] Victor Costan and Srinivas Devadas. Intel sgx explained. IACR Cryptology ePrint Archive, 2016(086):1–118, 2016.

[7] Ben Fisch, Dhinakaran Vinayagamurthy, Dan Boneh, and Sergey Gorbunov. IRON: functional encryption using intel SGX. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, pages 765–782, 2017.

[8] Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In Advances in Cryptology-CRYPTO 1999, pages 537–554. Springer, 1999.

[9] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In ACM CCS 2006, pages 89–98. ACM, 2006.

[10] Jinguang Han, Willy Susilo, Yi Mu, Jianying Zhou, and Man Ho Allen Au. Improving privacy and security in decentralized ciphertext-policy attribute-based encryption. IEEE transactions on information forensics and security, 10(3):665–678, 2015.

[11] Christofer Hoff. Cloud computing security: From ddos (distributed denial of service) to edos (economic denial of sustainability). http://www. rationalsurvivability.com/blog/?p=66.

[12] Joseph Idziorek, Mark Tannian, and Doug Jacobson. Attribution of fraudulent resource consumption in the cloud. In IEEE CLOUD 2012, pages 99–106. IEEE, 2012.

[13] Simon Johnson, Vinnie Scarlata, Carlos Rozas, Ernie Brickell, and Frank Mckeen. Intel R software guard extensions: Epid provision-ing and attestation services. White Paper, 1:1–10, 2016.

[14] Sangho Lee, Ming-Wei Shih, Prasun Gera, Taesoo Kim, Hyesoon Kim, and Marcus Peinado. Inferring fine-grained control flow inside sgx enclaves with branch shadowing. In 26th USENIX Security Symposium, USENIX Security, pages 16–18, 2017.

[15] Jiguo Li, Xiaonan Lin, Yichen Zhang, and Jinguang Han. Ksfoabe: outsourced attribute-based encryption with keyword search functionforcloudstorage. IEEETransactionsonServicesComputing, 10(5):715–725, 2017.

[16] Jiguo Li, Yao Wang, Yichen Zhang, and Jinguang Han. Full verifiability for outsourced decryption in attribute based encryption. IEEE Transactions on Services Computing, DOI: 10.1109/TSC.2017.2710190, 2017.

[17] Wei Li, Kaiping Xue, Yingjie Xue, and Jianan Hong. Tmacs: A robust and verifiable threshold multi-authority access control system in public cloud storage. IEEE Transactions on parallel and distributed systems, 27(5):1484–1496, 2016.

[18] Ben Lynn et al. The pairing-based cryptography library. Internet: crypto. stanford. edu/pbc/[Mar. 27, 2013], 2006.

[19] Frank McKeen, Ilya Alexandrovich, Alex Berenzon, Carlos V. Rozas, Hisham Shafi, Vedvyas Shanbhogue, and Uday R. Savagaonkar. Innovative instructions and software model for isolated execution. In HASP@ISCA 2013, page 10, 2013.

[20] Antonis Michalas. The lord of the shares: combining attribute based encryption and searchable encryption for flexible data sharing. In SAC 2019, pages 146–155, 2019.

[21] Jianting Ning, Zhenfu Cao, Xiaolei Dong, Kaitai Liang, Hui Ma, and Lifei Wei. Auditable σ-time outsourced attribute-based encryption for access control in cloud computing. IEEE Transactions on Information Forensics and Security, 13(1):94–105, 2018.

[22] JiantingNing,ZhenfuCao,XiaoleiDong,andLifeiWei. White-box traceable CP-ABE for cloud storage service: How to catch people leaking their access credentials effectively. IEEE Transactions on Dependable and Secure Computing, 15(5):883–897, 2018.

[23] Jianting Ning, Zhenfu Cao, Xiaolei Dong, Lifei Wei, and Xiaodong Lin. Large universe ciphertext-policy attribute-based encryption with white-box traceability. In Computer Security-ESORICS 2014, pages 55–72. Springer, 2014.

[24] Jianting Ning, Xiaolei Dong, Zhenfu Cao, and Lifei Wei. Accountable authority ciphertext-policy attribute-based encryption with white-box traceability and public auditing in the cloud. In Computer Security–ESORICS 2015, pages 270–289. Springer, 2015.

[25] Jianting Ning, Xiaolei Dong, Zhenfu Cao, Lifei Wei, and Xiaodong Lin. White-box traceable ciphertext-policy attribute-based encryption supporting flexible attributes. IEEETransactions on Information Forensics and Security, 10(6):1274–1288, 2015.