Data Security in Cloud Storage Services

Mai Mansour Dahshan
Department of Computer Science and Engineering
The American University in Cairo
Cairo, Egypt
mdahshan@aucegypt.edu

Sherif ElKassass
Department of Computer Science and Engineering
The American University in Cairo
Cairo, Egypt
sherif@aucegypt.edu

***Abstract*— Cloud storage services have changed the way used to manage and interact with data outsourced to public domains. With these services, multiple subscribers can collaboratively work and share outsourced data without any concerns about their data consistency, availability and reliability. Examples of these services include Dropbox, Box.net, UbuntuOne or JungleDisk. Although these cloud storage services offer seductive features, many customers are not rushing to move their data into these services. Since data stored in these services is under the control of service providers which makes it more susceptible to security risks. Therefore, using cloud storage services for storing users data depends mainly on whether it is sufficiently secure or not. From the way cloud storage services are constructed, we can notice that these storage services don't provide users with sufficient levels of security leading to an inherent risk on users' data from external and internal attacks. To deal with these security problems, this paper proposes a novel data sharing mechanism that simultaneously achieves data confidentiality, fine-grained access control on encrypted data and user revocation by combining ciphertext policy attribute-based encryption (CP-ABE), and proxy re-encryption (PRE).**

Keywords-Secure Storage; Cloud Computing; Proxy Re-encryption; Ciphertext Policy Attribute Based Encryption.

I. INTRODUCTION

Cloud storage is a newly developed concept in the field of cloud computation. It can be defined as a system that is composed of cluster, grid and distributed file systems that using application software coordinates a variety of different type's storage devices together to provide data storage and access service. Cloud storage allows users to outsource their data that has been managed internally within the organization or by individual users to the cloud. By doing so, users eliminate the concerns associated with the installation of the complex underlying hardware, save increasing high cost in data center management and alleviate the responsibilities of its maintenance [1]. Although cloud storage services are offering this number of benefits, they are facing many challenges for securing data in public clouds, which are generally beyond the same trusted domain as data owners.

Challenges associated with cloud storage services are unique ones because all of the involved entities (i.e., Cloud Service Provider (CSP) as Dropbox and subscribers seeking access to the outsourced data) can behave maliciously. CSPs, which provision the outsourced data, can assist unauthorized subscribers to gain illegal access to data or learn about user's confidential information leading to potential loss of privacy. On the other hand, subscribers of CSPs can utilize data sharing and collaborative functionalities of a cloud storage service, complimented with malicious intent of CSP, to compromise privacy of the outsourced data. In addition, most of these subscribers are unaware about the security measures adopted by a CSP, how often they are evaluated, and how well these security measures conform to standards and government regulations [2][3][4].

Importing users' data into cloud storage services (Dropbox, box, SpiderOak, etc.) can face at least one of the following threats. First, service operators can steal users' data and credentials because they have the capacity and the authority to get access to users' data easily. Second, they can authorize other users to access users' data. Last but not least, unlike data stored in users' personal computer, the data stored in the cloud is not isolated from other people's data. Therefore, the risk of data being attacked by the cloud increases.

The rest of the paper is organized as follows: Section 2 gives an insight into the problem. Section 3 reviews some related work that has been done to solve the addressed problem in cloud security. Section 4 presents our solution. The paper is concluded in Section 5.

II. PROBLEM STATMENT

A recent security flaw in the Dropbox authentication mechanism [5] begins the debate about whether cloud storage services are sufficiently secure to store sensitive data or not. A recent research [6] about Dropbox has shown that it suffers from three types of attacks which are hash value manipulation attack, stolen host id attack and direct download attack. Moreover, another cloud storage service as Box may not encrypt user files via Secure Sockets Layer (SSL) during transfer to/from Box and may not encrypt data within Box servers [7]. Even in the more secure storage service, SpiderOak, user's data is encrypted with his own private encryption key and his password which can make it inaccessible in case of password loss [8]. Furthermore, Hu et al. [9] evaluated four cloud storage systems: Mozy, Carbonite, Dropbox, and CrashPlan. After the evaluation, it was found out that none of these systems can provide any guarantees for data integrity, availability, or even confidentiality. Motivated by these limitations, we need to design secure cloud storage architecture. Such architecture aims to encrypt the data that will be uploaded into cloud and protect the keys used for encryption. More precisely, such architecture should provide:

 ISBN: 978-1-61208-338-4

1) Confidentiality: encryption of data before uploading it to the public cloud and decryption of data after downloading from the cloud.
2) Secure data sharing: only authorized users have access to data.

III. RELATED WORK

Public cloud storage services interact with their customers either through web interface, Application Programming Interface (API) or proprietary software clients. Therefore, these services allow the users to share and synchronize data files without any direct interaction between users or knowledge about data encryption, access control polices and key management. In addition, in file sharing, the users use the web interface to share data subscribers and non-subscribers according to certain privileges. This implies that the cloud storage services are responsible for data encryption, key management, file sharing, and synchronization which make it vulnerable to all of the above threats. Therefore, we need to a secure data in cloud storage service by enforcing data confidentiality and access control to outsourced data.

A. Data Confidentiality

Current cloud storage services try to secure user's data by encrypting them either on server side or client side. In server side encryption as it is the case in dropbox, the data owner relies on the service for securing its data; however, this solution isn't feasible for two reasons. First, the user will send his plaintext to service which exposes it to internal attacks where the attacker can exploit vulnerabilities of servers to achieve user's data. In other words, user's data, in addition to, encryption keys are stored at provider's servers. Second, there is no guarantee that the service will encrypt the data before uploading it to the cloud [10].

On the other hand, in client side encryption as it is the case in Wuala [11], the service encrypts user's data locally before it is uploaded to the cloud. Although client side encryption appears to be good method for securing users' data, it isn't efficient to do so. Since the keys involved in the process of encryption are managed by software manner. Moreover, these cloud storage services may be exposed to the following threats: a) key disclosure: the client software uses the decryption key stored on user machine to decrypt the encrypted data send from the cloud storage provider to obtain the clear text. The client software might send this key to the provider or some other unauthorized parties; b) Manipulated file content: since these cloud services support public key cryptography, the public keys of the users are known by some parties, including the provider. Server software may encrypt a malicious content making use of user's public key. The user can decrypt this content without detecting the fraud. This usually takes place because the data is usually not signed; and c) the most dangerous threat is a secret agent working at the provider. This agent may be able to manipulate the client software by injecting a malware in the customer's system [12].

B. Fine Grained Access Control

One of the most challenging issues in current cloud-based file sharing service is the enforcement of access control policies and the support of policies updates. However, current cloud storage services separate the roles of the data owner from the CSP, and the data owner does not interact directly with data users for the purpose of data access service, which makes the data access control in cloud storage services a challenging issue. Moreover, the current deployment model of cloud storage services cannot be fully trusted by data owners; as a result, traditional server-based access control methods are no longer applicable to cloud storage systems.

To prevent the un-trusted servers from accessing sensitive data in a traditional server-based system, traditional methods usually encrypt files by using the symmetric encryption approach with content keys and then use every user's public key to encrypt the content keys and only users holding valid keys can access the data. These methods require complicated key management schemes and the data owners have to stay online all the time to deliver the keys to new user in the system. Moreover, these methods incur high storage overhead on the server, because the server should store multiple encrypted copies of the same data for users with different keys [13] [14]. In addition, these methods [15] [16] [17] deliver the key management and distribution from the data owners to the remote server under the assumption that the server is trusted or semi-trusted. However, the server cannot be trusted by the data owners in cloud storage systems and thus these methods cannot be applied to access control for cloud storage systems[18][19].

Attribute-based encryption (ABE) [20] is regarded as one of the most suitable technologies for realizing a fine-grained attribute-based access control mechanism. Since its introduction, two complementary schemes have been proposed, which are: key-policy ABE (KP-ABE) [21] and CP-ABE [22]. In a KP-ABE scheme, the ciphertext is defined by a set of attributes; while the secret keys of the user are associated with an access policy (access structure).A user can decrypt the ciphertext, if and only if he has the required secret keys corresponding to attributes listed in the ciphertext. As a result, the encryptor does not have entire control over the encryption policy because the encryption policy is described in the keys. Therefore, the encryptor has to trust the key generators for issuing correct keys for authorized users. On other hand, in a CP-ABE scheme, the ciphertext is associated with an access policy (access structure); while the secret keys of the user are defined by a set of attributes. A user can decrypt the ciphertext, if and only if his attributes satisfy the access policy. Therefore, it is more convenient for use in the cloud environment, because the encryptor holds the ultimate authority about the encryption policy. Moreover, in CP-ABE schemes, the access policy checking is implicitly conducted inside the cryptography. That is, there is no one to explicitly evaluate the policies and make decisions on whether allows the user to access the data [22][23].

 ISBN: 978-1-61208-338-4

C. Revocation

In most data sharing services, users may join and leave the system frequently. This requires a periodical re-encryption of data, and regeneration of new secret keys to remaining authorized users. However, this traditional revocation scheme isn't applicable in cloud sharing services that have high turnover rate. Therefore, Pirretti et al. [24] introduces the first revocation scheme for ABE in which the attributes are extended with expiration dates. An improvement to this scheme [22] issues a single key with some expiration dates rather than a separate key for every time period before it. However, these methods aren't able to achieve user revocation in a timely fashion. They can just disable a user secret key at a designated time, but are not able to revoke a user attribute/key on the ad hoc basis. A better solution can be achieved by delegating the re-encryption and key generation to a third party. This third party has the capabilities to execute these computational intensive tasks, e.g., re-encryption, while leaking the least information. Proxy re- encryption [25][26] is a good choice, where a semi-trusted proxy is able to convert a ciphertext that can be decrypted by a user into another ciphertext that can be decrypted by another, without knowing the underlying data and user secret keys.

IV. OUR SOLUTION

Our goal is to design a secure cloud storage service that ban the cloud from getting access to owner's plaintext or credentials, perform user's revocation without re-encrypting the affected files. In order to achieve these goals, we utilize and uniquely combine the following advanced cryptographic techniques: CP-ABE and PRE. Particularly, the proposed service transfers the trust from the cloud to a trusted third party (TTP) service. Since the currently deployed encryption services in either the cloud or inside client side cloud storage services are vulnerable to security attacks, we address these vulnerabilities by using a TTP service. This TTP service has encryption/decryption service that can be employed either locally or on top of the cloud storage. Since not all data offers the same value and not all require the same degree of protection even if it is encrypted locally on user ma-chine. Therefore, the service offers different encryption algorithms according to data's severity. For achieving data confidentiality against unauthorized users, the TTP service collaborates with an Attribute Authority (AA) through CP-ABE to achieve fine grained access control. Last but not least, PRE is used to issue different encryption key with each revocation to prevent revoked users from access the data.

Our main contributions are: 1) to design trusted third party service that enables users to share data over any web-based cloud storage platform while data security is preserved. This service protects the confidentiality of the communicated data and it can be employed locally or remotely;2) to present a CP-ABE scheme which allows users to share data between owners and users while maintaining fine grained access control scheme;3)to propose an efficient revocation scheme for CP-ABE scheme through PRE scheme. The proposed scheme tries to resolve the flaw in the granting pattern of most PRE that is employed in combination with CP-ABE. In addition, the scheme shall delegation most of computational tasks to CSP.

A. Models and Assumptions

- The cloud severs are honest and curious, which means that the cloud administrators can be interested in viewing user's content, but cannot modify it.
- Neither data owners nor users will be always online. They come online only when necessary.
- Legitimate users behave honestly, by which we mean that they never share their decryption key with the revoked users.
- All communications between users/clouds are secured by SSL/TLS protocol.

B. Definition of System Model and Framework

The system consists of the following five entities:

AA is an independent attribute authority that is responsible for issuing, revoking and updating user's attributes according to their role or identity in its domain. The authority computes a system-wide public key that is used for all operations within the system, and master key at the initialization phase in order to generate private keys for data users.

CSP is a semi-trusted entity that includes a proxy server. It is responsible for providing data storage service (i.e., Backend Storage Servers). Proxy servers are servers that are always available for providing various types of data services (i.e., proxy re-encryption technique).

TTP is an independent entity that is trusted by all other system components, and has expertise and capabilities to perform extensive tasks. The trusted third party contains two services for ensuring data confidentiality: data encryption service and data decryption service. The data encryption service is in charge of encrypting users' data. It doesn't keep any data after the encryption. On the other hand, the data decryption service only decrypts the data. In addition, it would not store any data at its end, it only stores keys. These keys are stored on hardware devices for better security.

Data owner encrypts the data with the help of TTP service (which could be local or remote). Then, the owner defines the access policies over a set of attributes

Data user has a global identity in the system with which he is entitled a set of attributes.

C. Ensure confidentiality of data

Since cloud storage services allow users to access data from anywhere and from any device. This means that cloud storage services serve two types of users: desktop users and mobile users. These users trust the cloud storage service by different levels. Specifically, there are three levels of trust to cloud storage service: full trust, partial trust, and no trust.

For these reasons, we offer three security mechanisms for ensuring data confidentiality for the three levels of trust.

- No trust: in this state the cloud users don't trust the cloud or any trusted third party for its data.

Therefore, cloud users shall employ the TTP on a private cloud.
A private cloud is solely owned by a single organization and managed internally or a trustworthy third party. As the private cloud is meant for a single organization, the threat of compromise of data with outside world is mitigated. Enforcement and management of security policy become easier as these things have to deal with a single organization.

- Partial trust: in this state the cloud users may trust any trusted third party that is employed on top of cloud server for its data.
- Full trust: in this state the cloud users trust the CSP for its data. As a result, the CSP handles all operations related to data confidentiality.

Not all data offer the same value and not all require the same degree of protection even if it is encrypted by locally on user machine. Therefore, any organization must adopt data classification schemes according to their level of confidentiality. Different encryption algorithms can be used to each data type according to its importance. In our scheme each trust level provides a user with three levels of data classification (low, average, high). Each level is associated with an encryption algorithm that is suitable for its security level so as to achieve better performance [27].

D. Secure data sharing

Secure outsourcing data to an un-trusted server has been studied for decades. Researchers have proposed many solutions to protect confidentiality and control the access to the outsourced data. In this paper, we combine proxy based encryption and CP-ABE [22] in an efficient way to achieve fine grained access control. In our method, we propose two layers of encryptions: owner's level encryption and access control's level encryption to achieve efficient data sharing.

To upload a file to the CSP, the TTP, that is employed either locally or remotely, encrypts the file with a symmetric encryption algorithm based on the sensitivity level selected from the owner to this file. This requires the first encryption level (inner layer). Next, it encrypts the data file symmetric encryption key (DEK) with a public key generated from the AA of CP-ABE based on users' credentials to produce the second level of encryption(outer layer). After that, the file is uploaded to CSP. The main contribution of our scheme is separation of the encrypted data from the access control. Therefore the first layer of encryption is for encrypting the data while the second one is for the access control. Therefore, any change in the access policies will only affect the outer layer (data access layer) without affecting the encrypted data. In addition, we allow the cloud to re-encrypt both data and attributes without disclosing owner's data, keys and attributes to any party. The only information disclosed to the cloud is the proxy keys provided by the TTP. These keys will be used to re-encrypt user's data and attributes.

When a new user wants to join the system, the data owner has to define the role of user and sends this information to the AA to generate a secret key based on user's role. The user in turn can use this secret key to access the data.

E. User Revocation

Users may join and leave the system frequently, leading to constant key re-generation and re-distribution through additional communication sessions to handle user revocation. In a highly scalable system composed of thousands of users, such events may occur at relatively high frequency. Researchers have proposed revocation by attaching an expiry date to the keys or introducing proxies [22] [28]. However, these approaches suffer from delay in revocation, increasing the size of ciphertext, or affecting (re-keying) all the users including both the revoked and non-revoked ones. In addition, most of the proxies have a deficiency in their granting pattern which is: “all or nothing”. If the proxy knew the proxy key from user A to user B, all A’s ciphertexts can be re- encrypted to ciphertexts of B. In other words, we have to fully trust the proxy because it has full control over the re-encryption keys. Therefore, we tried to handle problem associated with proxy server with the help of [29].

Whenever a user revocation take place, the AA just generates proxy re-encryption keys. However, it will not be sent them directly to the proxy. Instead, every time a revocation takes place, AA generates a one-time re-encryption key for this session (revocation event) and sends it to the proxy. The one-time key is a randomization of the original re-encryption key which can be used re-encrypt data in the same session. Therefore, the proxy cannot re-encryption any new data with previous re-encryption keys generated in the previous session.

V. CONCLUSION AND FUTURE WORK

In this paper, we defined a new framework for data security in cloud storage services. Through this framework, we were able to achieve data confidentiality and fine grained access control. In addition, our scheme was able to shift most of the extensive computation load to the cloud as data re-encryption to the cloud. We also proposed a technique of flexible revocation that enables owners to revoke users with less computational requirements and avoids collusion between the proxy and the users. Our future work is to evaluate this system by implementing the entire architecture nd testing its behavior in order to prove its efficiency.

REFERENCES

[1] M. Armbrust et al., “Above the clouds: A Berkeley view of cloud computing,” EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS- 2009, Feb. 2009.

[2] M. S. Varshini, “An Improved Security Enabled Distribution of Protected Cloud Storage Services by Zero- Knowledge Proof based on RSA Assumption,” International Journal of Computer Applications, vol. 40, no. 5, Feb. 2012, pp. 18–22.

[3] W. Jansen and T. Grance, “Guidelines on security and privacy in public cloud computing,” NIST Special Publication, Dec. 2011, pp. 800–144.

[4] Cloud security appliance: Security guidelines for critical areas of focus in cloud computing.[Online]. Available:

https://cloudsecurityalliance.org/guidance/csaguide.v3.0.pdf [retrieved: Feb. 2014]

[5] "Dropbox authentication: insecure by design", Derek Newton, April 7, 2011, [Online]. Available: http://dereknewton.com/2011/04/dropbox-authentication-static-host-ids/ [retrieved: Jan. 2014]

[6] M. Mulazzani, S. Schrittwieser, M. Leithner, M. Huber, and E. Weippl, " Dark clouds on the horizon: Using cloud storage as attack vector and online slack space," Proc. of the 20th USENIX Conference on Security, Aug. 2011, pp. 5-5.

[7] http://techlogon.com/2012/03/09/box-com-security-issues-for-personal-accounts/ [retrieved: Mar. 2014]

[8] https://spideroak.com/ [retrieved: Feb. 2014]

[9] W. Hu, T. Yang, and J. N. Matthews, "The good, the bad and the ugly of consumer cloud storage," ACM SIGOPS Operating Systems Review, Jul. 2010, pp. 110–115.

[10] B. Chacos, "How to encrypt your cloud storage for free," PCWorld, Sep 25, 2012, [Online]. Available:http://www.pcworld.com/article/2010296/how-to-encrypt-your-cloud-storage-for-free.html [retrieved: 3, 2014]

[11] "Wuala," [Online]. Available: http://www.wuala.com (last accessed 7/1/2014)

[12] M. Borgmann et al, "On the Security of Cloud Storage Services," Fraunhofer Institute for Secure Information Technology SIT, March. 2012, pp. 44-47, [Online]. Available: http://www.sit.fraunhofer.de/en/cloudstudy.html [retrieved: Mar. 2014]

[13] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, “Plutus: Scalable secure file sharing on untrusted storage,” Proc. of the 2nd USENIX Conference on File and Storage Technologies. Berkeley, CA, USA, USENIX Association, Mar. 2003, pp. 29–42.

[14] E. Goh, H. Shacham, N. Modadugu, and D. Boneh, “SiRiUS: Securing remote untrusted storage,” Proc. of the Tenth Network and Distributed System Security (NDSS) Symposium, Citeseer, Feb. 2003, pp.131–145.

[15] D. Naor, M. Naor, and J. Lotspiech, “Revocation and tracing schemes for stateless receivers,” 21st Annual International in Advances in Cryptology–CRYPTO 2001, Springer, Aug. 2001, pp. 41–62.

[16] S. D. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, “Over-encryption: management of access control evolution on outsourced data,” Proc. of the 33rd international conference on Very large data bases, Sep. 2007, pp. 123–134, 2007.

[17] S. D. C. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, “A data outsourcing architecture combining cryptography and access control,” Proc. of the 2007 ACM workshop on Computer security architecture, CSAW ’07. New York, NY, USA, ACM, Oct. 2007, pp. 63–69.

[18] A. Sahai, and B.. Waters, "Fuzzy Identity-based Encryption," Proc. of the 24th annual international conference on Theory and Applications of Cryptographic Techniques (EUROCRYPT'05),Springer, May. 2005, pp. 457–473.

[19] W. Wang, Z. Li, R. Owens, and B. Bhargava, “Secure and efficient access to outsourced data,” Proc. of the 2009 ACM workshop on Cloud computing security, ACM, Nov. 2009, pp. 55–66.

[20] "Mozy," [Online]. Available: http://www.mozy.com [retrieved: Feb. 2014]

[21] R. Ostrovsky, A. Sahai, and B. Waters," Attribute-based Encryption with Non-monotonic Access Structures," Proc. of the 14th ACM conference on Computer and communications security (CCS 2007), Alexandria,VA, USA, Oct. 2007, pp. 195–203.

[22] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-policy attribute based encryption,” IEEE Symposium on Security and Privacy, May. 2007, pp. 321–334

[23] B. Waters, “Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," Proc. of the 4th International Conference on Practice and Theory in Public Key Cryptography (PKC’11), Springer, Mar. 2011, pp. 53–70.

[24] M. Pirretti, P. Traynor , P. Mcdaniel, and B. Waters "Secure attribute-based systems," Proc. of the 13th ACM Conf. on Computer and Communications Security. New York, ACM Press,Oct. 2006, pp. 99-112.

[25] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," Proc. of International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT), May. 1998, pp. 127–144, 1998.

[26] M. Green and G. Ateniese, "Identity-based proxy re-encryption," Proc. of the 5th international conference on Applied Cryptography and Network Security, Sep. 2007, pp. 288–306.

[27] H. Chuang, S. Li, K. Huang, and Y.Kuo, "An effective privacy protection scheme for cloud computing," 2011 13th International Conference on Advanced Communication Technology (ICACT) , Feb. 2011, pp.260-265.

[28] S. Tu, S. Niu, H. Li, Y. Xiao-ming, and M. Li, “Fine-grained Access Control and Revocation for Sharing Data on Clouds,” 2012 IEEE 26th International Parallel and Distributed Processing Symposium Workshops & PhD Forum , May. 2012, pp. 2146–2155.

[29] X. Wu, L. Xu, and X. Zhang. " CL-PRE: a certificateless proxy re-encryption scheme for secure data sharing with public cloud," Proc. of the 7th ACM Symposium on Information, Computer and Communications Security (ASIACCS '12), May. 2012, New York, NY, USA, ACM, pp. 87-88.