# Efficient Revocation for Multi-Authority Cloud Storage Systems

Noor U Sabha, Sriraksha T A, Shivaraj Kumar T H
*Department of computer science and engineering, C Byregowda Institute of Technology*
*Kolar, Karnataka, India*

***Abstract***

*Due to the high volume and velocity of big data, it is an effective option to store big data in the cloud, as the cloud has capabilities of storing big data and processing high volume of user access requests. Attribute-Based Encryption (ABE) is a promising technique to ensure the end-to-end security of big data in the cloud. However, the policy updating has always been a challenging issue when ABE is used to construct access control schemes. A trivial implementation is to let data owners retrieve the data and re-encrypt it under the new access policy, and then send it back to the cloud. This method, however, incurs a high communication overhead and heavy computation burden on data owners. A novel scheme is proposed that enable efficient access control with dynamic policy updating for big data in the cloud. Developing an outsourced policy updating method for ABE systems is focused. This method can avoid the transmission of encrypted data and minimize the computation work of data owners, by making use of the previously encrypted data with old access policies. Policy updating algorithms is proposed for different types of access policies. An efficient and secure method is proposed that allows data owner to check whether the cloud server has updated the ciphertexts correctly***.**

**Keywords**- *Attribute-based encryption; multi-authority cloud storage; attribute-level revocation; user-level revocation*

## I. INTRODUCTION

Big data refers to high volume, high velocity, and/or high variety information assets that require new forms of processing to enable enhanced decision making, insight discovery and process optimization. Due to its high volume and complexity, it becomes difficult to process big data using on-hand database management tools. When hosting big data into the cloud, the data security becomes a major concern as cloud servers cannot be fully trusted by data owners. As the name would itself suggest, big data is an enormous or huge data-set, with a massive and complex volume so as to make it extremely difficult to process in the way traditional datasets are being managed as of today. The huge dataset pose excessive challenges in terms of analysing, capturing, storing, sharing, visualizing, presenting and securing, as it is unwieldy. It can provide low-cost, high-quality, flexible and scalable services to users. In particular, cloud computing realizes the pay-on-demand environment in which various resources are made available to users as they pay for what they need. Cloud storage is one of the most fundamental services[1], which enables the data owners to host their data in the cloud and through cloud servers to provide the data access to the data consumers (users). However, it is the semi-trusted cloud service providers (CSPs) that maintain and operate the outsourced data in this storage pattern [2][3].

To prevent the unauthorized entities from accessing the sensitive data, an intuitional solution is to encrypt data and then upload the encrypted data into the cloud [6][7].Nevertheless,the traditional public key encryption and identity based encryption (IBE) [8] cannot be directly adopted. The reason is that they only ensure the encrypted data can be decrypted by a single known user, such that it will decrease the flexibility and scalability of data access control. Attributed-based encryption (ABE) proposed by Sahai and Waters in [9], can be viewed as the generalization of IBE [8]. Decryption is possible if and only if the attributes of cipher text or secret key satisfy the access policy. Goval et al. [10] formulated two complimentary forms of ABE: key policy ABE (KP-ABE) and cipher text-policy ABE (CP-ABE).

In KP-ABE, user’s secret key is associated with an access policy and each cipher text is labeled with a set of attributes; while in CP-ABE, each cipher text is associated with an access policy and user’s secret key is labeled with a set of attributes. Compared with KP-ABE, CP-ABE is more suitable for the cloud-based data access control since it enables the data owner to enforce the access policy on outsourced data. However, there remains several challenges to the application of CP-ABE in cloud-based data access control. On one hand, there is only one attribute authority (AA) in the system responsible for attribute management and key distribution [11], [12], [13].

## II. SYSTEM MODEL

As shown in Figure. 1, consists of five kinds of entities: CA, AAs, data owners, users and CSP.

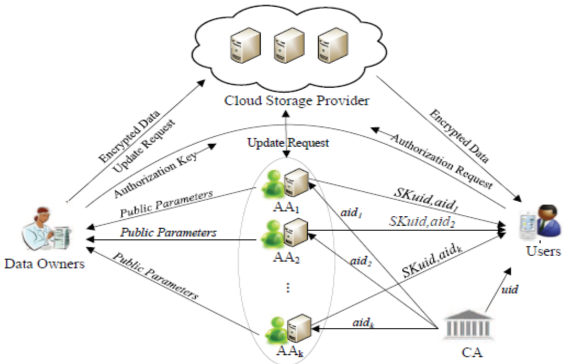

**Figure 1: Efficient Revocation For Multi-Authority Cloud Storage Systems**

The CA sets up the system and responses the registration requests from all the AAs and users. However, the CA is not involved into any attribute-related management. Each AA administers a distinct attribute domain and generates a pair of public/secret key for each attribute in this attribute domain. Without any doubt, each attribute is only managed by a single AA. Once receiving the request of attribute registration from a user, the AA generates the corresponding attribute secret keys for this user. Additionally, each AA is responsible to execute the attribute revocation of users.

Before uploading a shared data to the cloud storage servers, the data owner defines an access policy and encrypts the data under this access policy. After that, the data owner sends the cipher text and its corresponding access policy to the CSP. Meanwhile, the data owner is responsible for issuing and revoking the user's authorization.

Each user is labeled with a set of attributes, besides a global unique identifier. In order to obtain the shared data, each user needs to request the attribute secret keys and authorization from AAs and data owner, respectively. Any user can download the ciphertext from the CSP. Only the authorized user who has the specific attributes can successfully recover the outsourced data. It becomes obvious that the CSP provides data storage service and enforces the process of ciphertext update. The ciphertext update occurs in the following two cases: (1) any of AAs revokes users' one or more attributes; (2) the data owner revokes one or more authorized users.

## III. FRAMEWORK

The framework of efficient revocation for multi-authority cloud storage systems consists of the following phases:

### *Phase 1: Initialization of System*

First, the CA generates some global public parameters for the system, and accepts both the AA registration and user registration. Then, each AA and data owner respectively generate the public parameters and secret information used throughout the execution of system.

### *Phase 2: Generation Secret Key and Authorization*

When a user submits a request of attribute registration to AA, the AA distributes the corresponding attribute secret keys to this user if his/her certificate is true. When a user submits an authorization request to data owner, the data owner generates the corresponding authorization key and delivers it to this user.

### *Phase 3: Data Encryption*

For each shared data, the data owner first defines an access policy, and then encrypts the data under this specified access policy. Thereafter, the data owner outsources this ciphertext to the CSP. The encryption operation will use a set of public keys from the involved AAs and the data owner's authorization secret key.

### *Phase 4: Data Decryption*

All the users in the system are allowed to query and download any interested ciphertexts from the CSP. A user is able to recover the outsourced data, only if this user holds the sufficient attribute secret keys with respect to access policy and authorization key with regard to outsourced data.

### *Phase 5: Attribute-level Revocation*

For attribute-level revocation, the AA who manages the revoked attribute, issues a new public key to this revoked attribute, and generates attribute update keys for non-revoked users and a set of ciphertext update components for CSP. Each non-revoked user who holds the revoked attribute will update the corresponding attribute secret key upon receiving the attribute update key. Based on the set of ciphertext update components, the ciphertexts associated with the revoked attribute will be updated by the CSP.

### *Phase 6: User-level Revocation*

In order to revoke a user's access privilege, the data owner generates a new authorization secret key used for authorization , a set of authorization update keys for non-revoked users and a set of ciphertext update components for ciphertext update. When receiving the authorization update key, each non-revoked user updates the authorization key and obtains the new version. All the involved ciphertexts will be updated by the CSP based on the set of ciphertext update components.

### *A. Security Assumptions and Threat Models*

- The CA is a full trusted party.
- Each AA is also trusted. But, any of AAs will never collude with users.
- The CSP is honest but curious, namely semi-trust. It will correctly execute all the prescribed

operations, but may try to decrypt the ciphertexts stored in the cloud servers by itself.

- Each user is dishonest, and may collude with others to obtain unauthorized access to data. Meanwhile, each user is not allowed to expose his/her attribute secret keys and authorization key to an adversary.

## IV. CONCLUSION

A new data access control scheme for multi-authority cloud storage systems this scheme provides two-factor protection mechanism to enhance the confidentiality of outsourced data. If a user wants to recover the outsourced data, this user is required to hold sufficient attribute secret keys with respect to the access policy and authorization key with regard to the outsourced data. In our proposed scheme, both the size of cipher text and the number of pairing operations in decryption are constant, which reduce the communication overhead and computation cost of the system. In addition, the proposed scheme provides the user-level revocation for data owner in attribute-based data access control systems. Extensive security analysis, performance comparisons and experimental results indicate that the proposed scheme is suitable to data access control for multi authority cloud storage systems.

## REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and Z. Matei. A view of cloud computing. Communications of the ACM, 53(4):50–58, 2010.

[2] K. Yang, X. Jia, K. Ren, B. Zhang, and R. Xie. DAC-MACS: Effective data access control for multi-authority cloud storage systems. IEEE Transactions on Information Forensics & Security, 8(11):2895–2903,2013.

[3] X. Chen, J. Li, X. Huang, J. Ma, and W. Lou. New publicly verifiable databases with efficient updates. IEEE Transactions on Dependable and Secure Computing, 12(5):546–556, 2015.

[4] K. Ren, C. Wang, and Q. Wang. Security challenges for the public cloud. IEEE Internet Computing, 16(1):69–73, 2012.

[5] S. Subashiniand V. Kavitha. A survey on security issues in service delivery models of cloud computing. Journal of Network and Computer Applications, 34(1):1 – 11, 2011.

[6] S. Kamara and K. Lauter. Cryptographic cloud storage. In Proceedings of the 1st Workshop on Real-Life Cryptographic Protocols and Standardization(RLCPS'2010), volume 6054 of Lecture Notes in Computer Science, pages 136–149, Berlin, Heidelberg, 2010. Springer-Verlag.

[7] X. Chen, J. Li, J. Ma, Q. Tang, and W. Lou. New algorithms for secure outsourcing of modular exponentiations. IEEE Transactions on Parallel and Distributed Systems, 25(9):2386–2396, 2014.

[8] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. In Advances in Cryptology-CRYPTO'2001, volume 2139 of Lecture Notes in Computer Science, pages 213–229, Berlin, Heidelberg, 2001. Springer-Verlag.

[9] A. Sahai and B. Waters. Fuzzy identity-based encryption. In Advances in Cryptology-EUROCRYPT'2005, volume 3494 of Lecture Notes in Computer Science, pages 457–473. Springer Heidelberg, 2005.

[10] V. Goyal, O. Pandey, A. Sahai, and B. Waters. Attribute-based encryption for fine-grained access control of encrypted data. In Proceedings of the 13th ACM Conference on Computer and Communications Security(CCS'2006), pages 89–98. ACM, 30 October - 3 November 2006.

[11] J. Hur and D. K. Noh.Attribute-based access control with efficient revocation in data outsourcing systems. IEEE Transactions on Parallel and Distributed Systems, 22(7):1214–1221, 2011.

[12] J. Lai, R. H. Deng, C. Guan, and J. Weng. Attribute-based encryption with verifiable outsourced decryption. IEEE Transactions on Information Forensics and Security, 8(8):1343–1354, 2013.

[13] K. Yang, X. Jia, and K. Ren. Attribute-based fine-grained access control with efficient revocation in cloud storage systems. In Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security(ASIACCS'2013), pages 523–528, New York, NY, USA, 2013. ACM.

[14] S. Yu, C. Wang, K. Ren, and W. Lou. Attribute based data sharing with attribute revocation. In Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security(ASIACCS'2010),pages 261–270, New York, NY, USA, 2010. ACM.

[15] J. Bethencourt, A. Sahai, and B. Waters. Ciphertext-policy attribute based encryption. In Proceedings of the 2007 IEEE Symposium on Security and Privacy(S&P'2007), pages 321–334. IEEE, 20-23 May 2007.