Indian Journal of Science and Technology, *Vol 8(9),340-345, May 2015*

ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
DOI: 10.17485/ijst/2015/v8iS9/65590

Secure Data Sharing in Scalable Mobile Cloud Environment using HABE with Re-Encryption

B. SaranyaDevi, S. Shruthi, S. ThivyaRajeswari, P. Shanthi* and A. Umamakeswari

Department of CSE, School of Computing, SASTRA University, Thanjavur – 613401, Tamil Nadu, India; saranya.devi079@gmail.com, shruthideep1902@gmail.com, divi1123@gmail.com, shanthip@cse.sastra.edu, aum@cse.sastra.edu

Abstract

As cloud is a semi-trusted third-party entity, sharing of confidential data may not be secure. To achieve security many cryptographic algorithms are adopted. For better security and scalability of sharing data over cloud from mobile devices, HABE along with Group Key mechanism is used. With this the data are stored securely in cloud and the system is scalable for group communication. To reduce computational overhead during revocation Proxy re-encryption technique is used. It also improves the performance of the mobile users by leveraging the process of regenerating the keys onto the server side.

Keywords: Cloud Security, Group Key Mechanism, HABE, Proxy Re-Encryption

1. Introduction

With the devices such as Smart phones and tablets equipped with 3G and WIFI technologies data in the cloud can be remotely accessed[1]. To inhibit security data is stored in encrypted format in the cloud. Numerous solutions are used to exchange encrypted data with a cloud provider in a secure manner. In traditional public key infrastructure such as RSA[2] for each user the data encryption and decryption is done by public key private key respectively. So if data is to be shared among thousands of users, thousands of public and private keys pairs are generated and maintained. To overcome this, Attribute Based Encryption[3] and Identity Based Encryption schemes[4] are used where the data is encrypted with single public key and user's attributes and user's identity. The users who have those attributes that matches the identity should decrypt the data. Users may join and leave the authorized user set frequently in group communications[5]. Since the number of user' accessing the secret data is not constant, scalability is a prime concern. Hierarchical Attribute Based Encryption is adopted to achieve better scalability[6]. During revocation new key is generated and the data is decrypted and again encrypted with new key. Data in cloud should be stored in encrypted form so that the cloud provider cannot access it. The encrypted data should ideally be transformed such that it may be unlocked with new keys, without an immediate decryption step that would allow the cloud provider to read the plaintext; this process is known as data re-encryption[7].

Constant generation of keys and redistribution occur at relatively high frequency in highly scalable systems but in mobile devices with limited memory and battery power, it is expensive and results in rapid battery drain, especially when transmitting[8]. So with this limitation of mobile devices, during revocation instead of regenerating and redistributing secret key, users are allowed to upgrade a common group key, in order to reduce the communication cost which results in higher efficiency. It is also taken into account that the computations done in mobile devices must also be minimized[9].

1.1 Attribute Based Encryption

Many cryptographically enforced access control methods provide increased protection in data

**Author for correspondence*

Doutsourcing[10,11]. Based on the attributes of user or information to be shared, Attribute based Encryption schemes define access policies for each user. Attributes are used to generate public key for encrypting the data and also as access policy for authorized access. The access policy can be categorized as either cipher text-policy or key policy. The cipher text-policy is the access structure on the cipher text and the key-policy is the access structure on the user's private key.

The key policy based ABE[12] scheme provide flexibility and fine grained access control but it is unsuitable for certain applications as the data owner could not choose who can decrypt the data except choosing a set of attributes to describe the data. The cipher text-policy based ABE scheme builds access policy on encrypted data[13]. It allows user to obtain access to encrypted data in the cloud based on the possession of certain attributes that satisfy an access structure defined in the cloud, rather than the key that must be distributed to all interested parties in advance. The requisite attributes are generated by a data owner in advance and he is responsible for generating the data to be shared, encrypting it and uploading it to the cloud. CP-ABE schemes demands data owner granting access permission through an access tree, which requires data owner's constant availability and maintaining large set of secret keys lacks flexibility and scalability. On one side attribute based encryption schemes are growing to achieve fine grained access control and on other side Identity based encryption schemes which are weightless algorithms on client side provides better authentication[14] are used to provide collision resistance. Hierarchical identity based an encryption scheme which is the generalization of IBE schemes that creates organizational hierarchy in generating user's identity[15].

1.2 HABE with Group Key Mechanism

To achieve flexibility in key generation CP-ABE and HIBE schemes are combined and Hierarchical Attribute based Encryption schemes are proposed[16]. The HABE imbibes the hierarchical generation of keys from HIBE systems and flexible access control from CP-ABE systems. Though HABE scheme provides fine grained access control and scalability, there is an overhead in number of keys updated during user revocation. To reduce the number of keys updated during revocation Group Key mechanism is implemented along with HABE. Re-encryption techniques are used to handle user revocation. The traditional approach is decrypting the cipher text and again encrypting it with the new key. The drawback of this approach is the data can be accessed in this intermediate process. To overcome this proxy re-encryption mechanism is used. Also re-encryption is done on cloud side; the data owner need not be available all the time for regenerating the keys.

In this paper, we propose an algorithm using HABE scheme with Group key mechanism to achieve performance by reducing the number of keys generated during user revocation. Proxy re-encryption technique is used, so that re-encryption techniques are done on cloud side which reduces computations on mobile devices to conserve battery power.

2. System Model

Our system involves 5 major roles:

- Root Authority (RA).
- Domain Authority (DA).
- Data Owner.
- Data User.
- Cloud Service Provider (CSP) as shown in Figure 1.

The RA and DA are the trusted authorities within the organization. They are responsible for key generation for the users in the organization. The android mobile users who are accessing our system using android application are the end users. These end users are classified as Data Owner and Data user. Data Owner is the person who stores the data in encrypted form in cloud and data users are people having rights to access that data. CSP provides perpetual data storage which can be accessed remotely over public internet. RA generates public parameters and public key for the system and Master keys for the next level DA. DA generates Master keys for their next level DAs and also secret keys and secret attributes for the end users. The keys generated are stored in the private cloud managed by admin.

An access policy is built based on attributes of the requesters who can access the data and also the data owner generates a pair of keys as Group Public Key (GPK) and Group Secret Key (GSK). The data owner encrypts the data using system public key, access policy and GPK and the encrypted data is stored in public cloud. In this model keys are stored in private cloud while encrypted data is stored in public cloud so for optimization purpose hybrid cloud is used[17].

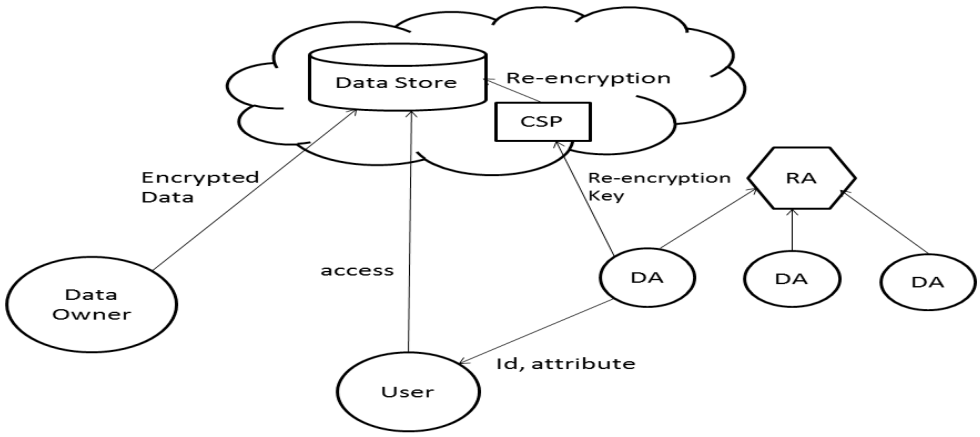

Figure 1. Architecture Diagram.

The data user can access that data and decrypt it using his secret key, secret attribute and GSK. During revocation of some user from that group, the access policy has to re-build and the data has to be re-encrypted in cloud. At that time the semi-trusted CSP may access the data. Instead of reforming the access policy, the group key is re-generated and given to the CSP and the data is re-encrypted using that key. In this case the CSP cannot access the data since it is dual encrypted using Public key and GPK and he is provided only with re-generated GPK. This new GSK is not known to the left user and hence he cannot access the data.

3. Proposed Algorithm

The HABE scheme we proposed involves HIBE and CP-ABE schemes along with Group Key mechanism.

3.1 HABE with Group Key Mechanism

The HABE scheme along with Group Key mechanism involves the following algorithms:

3.1.1 Setup (K)

This algorithm is carried out by RA by taking a security parameter K as input and generates System Public Key parameters (params), System Public Key (PK) and root Master Key (MK).

3.1.2 Domain Creation (param,MK_i,PK_{i+1})

This algorithm is carried out by RA or DA. Based on public parameters, Master Key of successor (RA or DA) and Public Key of current DA, Master Key (MK_i) is generated.

3.1.3 User Creation (params, MK_i, PK_{user}, PK_{attr})

This algorithm is carried out by DA whenever a new user enters the system. It takes public parameters, DA's Master Key, User public key and user public attribute as input and generates secret key ($\mathbf{PK_{user}}$) and secret attribute (PK_{attr}) for the user.

3.1.4 Group Creation (A, $\{PK_a \mid a \in A\}$)

This algorithm is carried out by the data owner when encrypting and storing the data into cloud. Based on access control policy and Public attributes of the users who can access the data, a pair of keys known as Group Public Key (GPK) and Group Secret Key (GSK) is generated.

3.1.5 Encrypt (params, M, PK, GPK)

The data owner encrypts the message M using public parameters (params), System Public Key (PK) and Group Public Key (GPK) and stores the Cipher Text (CT) into the cloud.

3.1.6 Decrypt (params, CT, GSK)

The data user can access the cloud and get CT. The message M is recovered from CT using the user's Secret Key, Secret attribute and Group Secret Key (GSK).

3.2 Revocation

During revocation group key is regenerated and message is re-encrypted. To do this, the algorithms are:

3.2.1 Key Regen

This algorithm is carried out by the DA of respective Data Owner. This outputs a new pair of Group Keys (GPK and GSK). This GPK is given to the CSP and GSK is sent to the valid users.

3.2.2 Re-Encrypt (CT, New GPK)

The CSP re-encrypts the CT using newly generated GPK such that the data users can access and decrypt the data using his secret parameters and newly generated GSK.

4. Security Analysis

The proposed system satisfies security criteria such as data confidentiality, scalability, fine-grained access control, user revocation and collusion resistant. The data to be shared is encrypted before uploading into cloud providing data confidentiality. The access rights of the users differ based on the access policy constructed. This provides fine-grained access control over the data. Any user can join the system at any time. This increase in number of users cannot affect the system's performance and efficiency. And also any user can leave the system at any time. At that time the user's access rights are revoked directly from the system such that the left user cannot access the data. This ensures that the system is more scalable and also provides efficient user revocation. Different users cannot collude by combining their attributes to decrypt the data. This ensures collusion resistant of the system. Our system is more resistive against adaptive chosen plaintext attack which is proved under BDH assumption and the random oracle model[18]. Since the authorized users can decrypt the data only if they possess secret key and group key and also if their attributes pass through the access policy, shows that HABE with group key mechanism is more secure with dual layer security.

5. Results

HABE with group key mechanism is compared with other encryption algorithms such as KP-ABE, CP-ABE and HABE without group key mechanism. Figure 2 shows that number of keys regenerated during revocation increases gradually as the number of users increases whereas fig 3 shows that the number of keys remain constant as 1 for any number of users. Table 1 also shows comparison of various parameters between these algorithms.

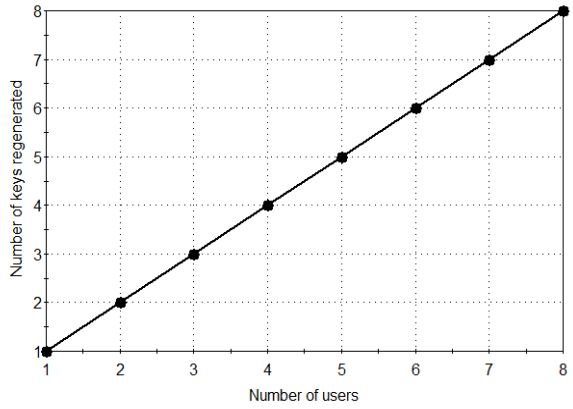

Figure 2. Key Regeneration during revocation in other systems.

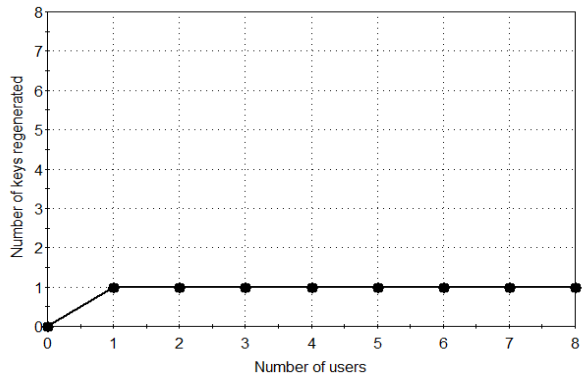

Figure 3. Key regeneration during revocation in HABE with group key mechanism.

6. Conclusion

The most challenging issue in outsourcing the data over cloud environment is security. Various cryptographic protocols are used in-order to provide security. The security in scalable environment is achieved by using HABE scheme where the number of users accessing the data is not constant. In this paper HABE with Group Key mechanism is implemented which provides scalability and reduced the number of keys generated during re-encryption. Use of Proxy Re-encryption scheme allows re-encryption process to be carried by CSP without decrypting the data which also reduces overhead to data owner. Overall the proposed system provides effective solution for sharing confidential data over cloud environment via mobile devices and also results in improved performance.

Table 1. Comparison of different attribute-based encryption algorithms

Characteristics	KP-ABE	CP-ABE	HABE (without group key)	HABE (without group key)
User Revocation	Regenerates key for each user	Regenerates key for each user	Regenerates key for each user	Regenerates single group key
Key Delegation	Data Owner	Attribute Authority	Domain Manager	Domain Manager
Parameters for decryption	Access subtree	User Id and User attribute	Access policy and regenerated secret key	Access Policy, Secret key and Generated group key

7. Future Work

At the time of decryption, the user needs to decrypt the data with both the secret key and group key. This incurs additional decryption cost. In future, a system with reduced decryption cost will be implemented.

8. References

1. Lee J-Y. A study on the use of secure data in cloud storage for collaboration. Indian Journal of Science and Technology. 2015 Mar; 8(S5):33–6.
2. Rivest RL, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Comm ACM. 1983 Jan; 26(1):96–9.
3. Hur J, Noh D. Attribute-based access control with efficient revocation in data outsourcing systems. IEEE Trans Parallel Distr Syst. 2011 Jul; 22(7):1214–21.
4. Sahai, Waters B. Fuzzy identity based encryption. Advances in Cryptology V EUROCRYPT of LNCS. 2005; 3494:457–73.
5. Naor D, Naor M, Lotspiech J. Revocation and tracing schemes for stateless receivers. Proceedings of International Cryptology Conference on Advances in Cryptology (CRYPTO '01); 2001. p. 41–62.
6. Lee CC, Chung PS, Hwang MS. A survey on attribute-based encryption schemes of access control in cloud environments. Int J Netw Secur. 2013 Jul; 15(4):231–40.
7. Liu Q, Wang G, Wu J. Clock-based proxy re-encryption scheme in unreliable clouds. Proceedings of 41st International Conference Parallel Processing Workshops (ICPPW); 2012 Sep. p. 304–5.
8. Balasubramanian N, Balasubramanian A, Venkataramani A. Energy consumption in mobile phones: A measurement study and implications for network applications. Proceedings of 9th ACM SIGCOMM Conference on Internet Measurement Conference (IMC '09); 2009. p. 280–93.
9. Yu S, Wang C, Ren K, Lou W. Achieving secure, scalable, and fine-grained data access control in cloud computing. Proceedings of IEEE INFOCOM '10; 2010. p. 534–42.
10. Vimercati S, Foresti S, Jajodia S, Paraboschi S, Samarati P. A data outsourcing architecture combining cryptography and access control. Proceedings of ACM Workshop Computer Security Architecture (CSAW '07); 2007 Nov. p. 63–9.
11. Ibraimi L, Petkovic M, Nikova S, Hartel P, Jonker W. Mediated ciphertext-policy attribute-based encryption and its application. Proceedings of International Workshop Information Security Applications (WISA '09); 2009. p. 309–23.
12. Goyal V, Pandey O, Sahai A, Waters B. Attribute-based encryption for fine-grained access control of encrypted data. Proceedings of the 13th ACM Conference on Computer and Communications Security; 2006. p. 89–98.
13. Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryption. Proceedings of IEEE Symposium on Security and Privacy; 2007. p. 321–34.

14. Rajathi, Saravanan N. A survey on secure storage in cloud computing. Indian Journal of Science and Technology. 2013 Apr; 6(4):4396–440.
15. Horwitz J, Lynn B. Towards hierarchical identity based encryption. In: Knudsen LR, editor. EUROCRYPT 2002 of LNCS. Springer-Verlag; 2002. p.466–81.
16. Wang G, Liu Q, Wu J. Hierarchical attribute-based encryption for fine-grained access control in cloud storage services. Proceedings of 17th ACM Conference on Computer and Communications Security (CCS'10); 2010. p. 735–7.
17. Sharifi AM, Amirgholipour SK, Alirezanejad M, Aski1 BS, Ghiami M. Availability challenge of cloud system under DDOS attack. Indian Journal of Science and Technology. 2012 Jun; 5(6):2933–7.
18. Gentry C, Silverberg A. Hierarchical ID-based cryptography. Proceedings of ASIACRYPT; 2002. p. 548–66.