Indian Journal of Science and Technology, Vol 10(34), DOI: 10.17485/ijst/2017/v10i34/114130, September 2017
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Improved Attribute based Encryption Scheme over Integrated Data Access Control Structures in Cloud Computing

Afreen Rafiq*, D. Sai Eswari and R. Deepthi

Department of Computer Science and Engineering, Sri Satya Sai University of Technology and Medical Sciences, Sehore – 466001, Madhya Pradesh, India;
afreenrafiq2@gmail.com, saieswari3@gmail.com, deepthideepu66@gmail.com

Abstract

The advancements made in the Information and Communication Technologies (ICT) has increased the number of social users. A new promising technology, named, cloud computing resolves most of the recent real-time issues. Data sharing is one of the most recent topics explored by the researchers. By sharing the common attributes, the data owners can elegantly solve the data redundancy, cost minimization/ maximization, security etc. This kind of sharing attributes are profound in healthcare and military applications. Prior works have explored the study of data sharing concepts, yet it fails to satisfy the data owner's requirements. In this paper, we have proposed an enhanced file hierarchy system in specific to attribute-based encryption schemes. Meanwhile, the file hierarchical system is controlled by data access control layer of cloud technologies. By devising the access control layer, we can efficiently arrange encrypted files in hierarchical model with reduced storage and time cost of encryption and decryption. Experimental analysis has shown the efficacy of our proposed scheme in terms of better achieved storage consumption with increased number of files and attributes.

Keywords: Attributes, Cloud Computing, Data Owner, Encryption, File Hierarchy Systems, Storage Consumption

1. Introduction

The recent advancements in the cloud technologies attract profound users, and thus a vast amount of information is being generated. Once the data is being outsourced, the users are unaware about their data i.e. lack of knowledge on data location, data misuse, data leakage etc. In order to achieve the best of cloud technologies, data security is the most vital part of the Information and Communications technologies[1]. The maintenance of control over the cloud data is the paramount to the success of the cloud. Generally, the cloud environment is subdivided into three categories, namely, public, private, and hybrid cloud. The sensitive information is shared among the cloud environments. Each environment possess a unique security features in it. Most of the organizations relied upon the third party service providers to facilitate proper service to their clients. But they lack to support visibility into the sensitive data. Data sharing is the primary layer over the sensitive data[2]. To protect the data from unauthorized users, Access control is the only solution persists to prevent the unauthorized access to the shared data.

The growth of network technology and mobile terminals has enhanced the concept of data sharing. The networks environments Facebook, Myspace, etc. has renowned for sharing their data in different forms. In addition to cloud has become a promising application for the growth of data sharing. Before sharing the data, the data has to encrypted and then outsourced to the cloud environment to eliminate the data leakage issue. In the perspective of cloud computing, when the user submits their data, a security parameter is considered for validating the users for upcoming actions[3]. Cloud Service Providers (CSP) is the main ingredient between cloud users and

*Author for correspondence

cloud environments. It offers an end-to-end service for their clients. Data owner encrypts the data and then upload to the cloud servers. When any user is annoyed of viewing the files, they submit their security parameter to the cloud service and then validated for viewing the files. Generally, the encrypted files are stored in hierarchical index structure at different access levels for better security and easy file retrieval process. By doing so, we can eliminate high storage costs and encryption time costs[4].

Let us consider healthcare environments i.e. Personal Health Records (PHR). In order to securely share their sensitive data to the cloud environment, the data owner subdivides their data into two parts, medical record m1 and medical record m2, where m1 contains details of personal like name, security number, address etc. and m2 contain sensitive details like test records, treatment protocols etc. Prior works like Ciphertext policy based Attribute based Encryption (CP-ABE) scheme is used for encrypting and providing different access policy to both m1 and m2 records[5]. By adopting this scheme, the healthcare environments protect their sensitive data. Yet, it fails to completely eradicate the data leakage issues. Thus, our research study focuses on efficient file hierarchical scheme for better healthcare applications in cloud environment.

The paper is unionized as follows: Section II presents the prior works; Section III depicts the problem formulation and its solution i.e proposed work; Section IV demonstrates the experimental evaluations and concludes in Section V.

2. Prior Works

This section depicts the basic details of Attribute Based Encryption and its versions like Ciphertext policy and Key policy.

2.1 Attribute Based Encryption (ABE)

Attribute based Encryption is developed by Sahai and Waters in 2005. The objective of their scheme is to provide better security and access control. Relied upon the user's attributes, the encryption and decryption process is done[6]. It falls under the category of public key encryption schemes. In this secret key and the ciphertext is created for the specified user's attributes. The decryption process is achieved only when the user submitted secret key is validated. Collusion resistance is crucial security feature of Attribute Based Encryption. An adversary that holds multiple keys should only be able to access data if at least one individual key grants access[7–9]. The problem with attribute based encryption (ABE) scheme is that data owner needs to use every authorized user's public key to encrypt data. The application of this scheme is restricted in the real environment because it uses the access of monotonic attributes to control user's access in the system.

2.2 Key Policy based Attribute based Encryption (KP-ABE)

The Key policy based Attribute based Encryption (KP-ABE) is introduced in 2006. The main characteristics of the KP-ABE scheme is the set of attributes used for encrypting data and access policy is constructed from user's private key[10]. If the attributes are validated, then the access structure of the user's private key is used for obtaining the message. In order to obtain access structures, it takes polynomial functions such as $q_x(0) = q_{parent}(x)$ (index(x)) where $q_{parent}(x)$ is the x's parent node and their file indexes. In the access structure model, from root node r to each node x, the file is searched. So q(0) is equal to the master key y, and the master key y is distributed among the user's private key component D which is corresponding to the leaf node. It contains four processes, namely[12,13]:

a) Setup (d): It chooses a set of random numbers from the finite field Z and makes the public key. Using the public key PK, the secret key (SK) is generated by the parties.
b) Key_Gen (Au_ KP, PK, SK): The authority Au-KP contain private key components for every node x in access structure. If the secret key and Au-KP are mutually verified, then the files are sent to users.
c) Encrypt (M, ACT, PK): Data owner picks any random numbers from Z and encrypts the message with its attributes ACT and then the encrypted data is formed.
d) Decrypt (CT,D): It inputs the encrypted data, user's private key, and nodes of the access structure in user's private key. If i is equal to the leaf node, and i is in the access structure of user's private key, it will call the decrypt node function.

This scheme involves user's private key to its access structure. Thus, it has to satisfy both private keys and access structure to obtain the message which consumes higher time. In some scenario, the user can access the data structure without using private keys (Figure 1).

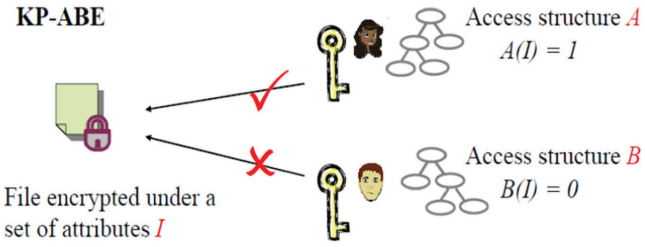

Figure 1. KP-ABE scheme- Workflow[11].

2.3 Ciphertext Policy based Attribute based encryption (CP-ABE)

Ciphertext policy based Attribute based Encryption (CP-ABE) is introduced by Bethencourt et al, 2007. It works similar to the key –policy based attribute based encryption. In this access control structure is used for creating the ciphertext. And a set of descriptive attributes are associated with the user's private key[14], and the access policy is built in the encrypted data. The access structure of the encrypted data is corresponding to the user's private key with a set of descriptive attributes. If a set of attributes in user's private key satisfies the access structure of the encrypted data, the data user can decrypt the encrypted data; if it cannot, the data user cannot obtain the message. It contains five processes, namely,

a) Setup: The authorities picks two random numbers from the finite field Z and the public key (PK) and master key MK are generated.
b) KeyGen (MK, Au): For each attribute and its random number, the user's private key is generated.
c) Encrypt (PK, M, ACT-CP): It is used for encrypting the messages. It encrypted with the access structure ACT-CP and the output is generated.
d) Delegate (D, Au'): It again takes the user's private keys and group of attributes to generate user's private keys.
e) Decrypt (CT, D): Once the encrypted data is received, the decryption process is executed. It makes use of Lagrange coefficient to compute the private key and access structure is accessed.

The CP-ABE builds the access structure in the encrypted data to choose the corresponding user's private key to decipher data. It improves the disadvantage of KP-ABE that the encrypted data cannot choose who can decrypt[15]. It can support the access control in the real environment. In addition, the user's private key is in this scheme, a combination of a set of attributes, so a user only uses this set of attributes to satisfy the access structure in the encrypted data (Figure 2).

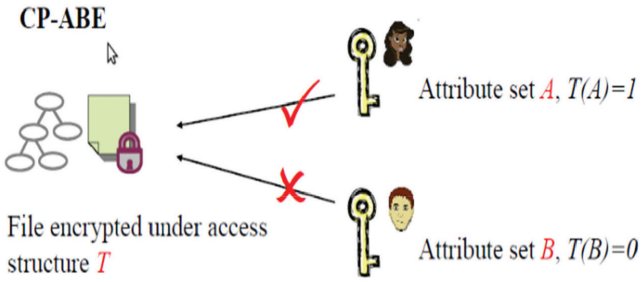

Figure 2. CP-ABE scheme –Workflow[15].

3. Improved and Efficient File Hierarchical Attribute based Encryption Scheme

This section discusses about an improved and efficient file hierarchical scheme. The proposed algorithm consists of four entities, namely, authority, Cloud service provider, data owner, and user. Let us consider data owner has files F with k access files. Let m be the medical records (m_1,……m_k) where m1 is the highest records and mk is the lowest records. If the user decrypts the m1, the other records can also decrypt. The following processes are explained as follows:

a) *Authority:* Authority is the trusted entity. Before accessing the cloud environment, the user should get enrolled with the cloud server. It takes the responsibility of Setup and KeyGen operations.
b) *Cloud service provider:* It is semi-trusted entity. It performs the sensitive results. It helps to store the sensitive data.
c) *Data owner:* It helps to store the data towards the cloud environment. The data is in ciphertext form and transformed to the cloud server.
d) *Data user:* It wants to access a large number of data in cloud system. The entity first downloads the corresponding ciphertext. Then it executes Decrypt operation of the proposed scheme (Figure 3).

The algorithm is explained as follows:

a) Setup (K)→(Params,MK_0):

- Let us pick random number MK_0 from finite field Z_q. Selecting the groups G_1 and G_2 of order q.
- Bilinear map e: G1* G1→G2 with random oracles H1: {0,1}→G1and H2: G2→{0,1}^{n}
- Hence, the params = (q, G_1, G_2, e, n,H_1,H_2) will be publically available with secret key, MK_0

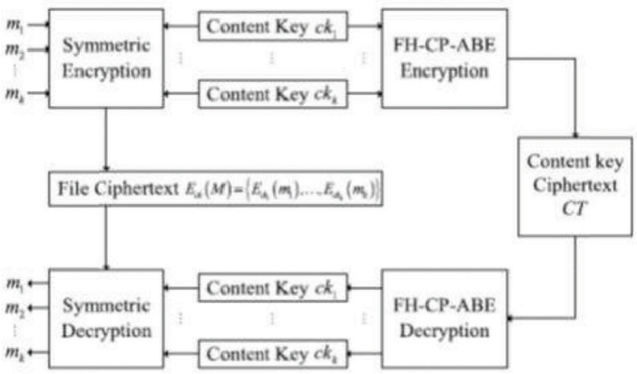

Figure 3. Proposed workflow.

b) *CreateDM (params,$M_{ki,Pki+1}$)-→(M_{ki+1}):*
- In order to generate the master key, select the random element mki+1from the finite element.
- Estimate SK_{i+1} =S_{ki}+$m_{ki}P_{i+1}$ where P_{i+1}= $H_1(Pk_{i+1})$

c) *Encrypt (params,A,(Pk_{aij}|1<i<N))→Ciphertext(CT):*
- Determine the DNF access control policy of $A = U_{i=1}^{N} CC=U_{i=1}^{N} U_{j=1}^{n_j} a_{ij}$ where N€Z^+ is the number of conjunctive clause in A.
- The sender computes the ciphertext CT=(A,Cf) where Cf =[U_0, U_{12}, ……, U_{ntN},U_N,V]

d) *Decrypt(Params,CT,Sk_{itl},u,{Sk_{it},u,a_{ij}|1≤j≥n_j}):*
- The user who satisfies the attributes CC_i can decrypt the message.
- Using XOR operation, the decryption process is as follows:

$$V \oplus H_2\left(\frac{\hat{e}(Q_0, n_A r P_1) \prod_{k=2}^{t_f} \hat{e}(Q_{t(k-1)}, n_A U_{tk}) \hat{e}(SK_{1t_1,u}, \frac{n_A}{n_t} U_t)}{\hat{e}(SK_{1t_1,u}, \frac{n_A}{n_t} U_t) \prod_{j=2}^{t_f} \hat{e}(U_{tj}, n_A Q_{t(j-1)})}\right)$$

$$V \oplus H_2(\hat{e}(Q_0, n_A r P_1))$$

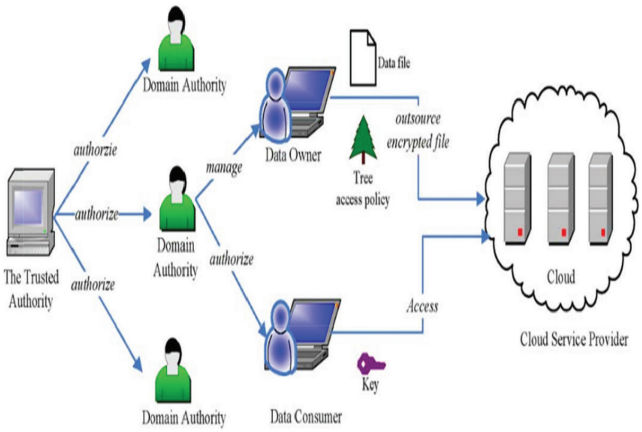

Figure 4. System architecture.

4. Performance Analysis

This section depicts the experimental analysis of our efficient file hierarchical attribute based encryption scheme. Consider a files F= (F_1,…..F_m) under authority who carry out Domain Granularity for every new user under different domain. Suppose the domain authority DA has a private key with some number of attributes. When DA wants to delegate some amount of the attributes, the cost produces linearly with the number of subsets to be assigned. The performance parameters analyzed are discussed as follows:

a) *Setup time:*

Every given number of attribute has linearly increases in cost-wise for its setup operations. From the given Figure 5, it is inferred our proposed work better than previous CP-ABE.

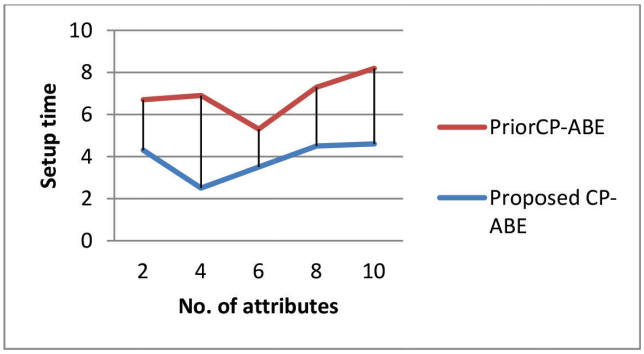

Figure 5. Setup time comparison.

a) *Key generation time:*

Generally, the cost of storage system is determined from its attributes generation and selection. According to attributes generation, the cost and time is determined for authorizing the users. From the Figure 6, it is inferred that our proposed CP-ABE decreases linearly with time varied attributes.

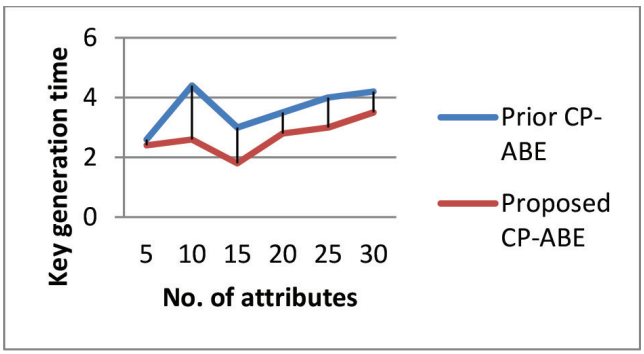

Figure 6. Key generation time.

c) Encryption Time:

In order to achieve the success of our proposed scheme, encryption time is the most significant factor. If the time taken for encryption is higher, the storage size is also linearly increases and degrades the performance of the file storing process. Henceforth, our proposed scheme employs lower time for encrypting the data which is shown in Figure 7.

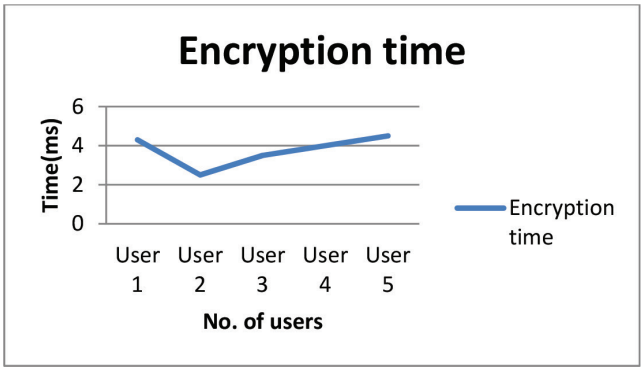

Figure 7. Encryption time.

d) Decryption Time:

The time taken for decryption operation relied upon the access tree structure. The time of decryption is different depending on the access tree and key structure. It assumes that there is just 1 subset with 40 attributes in the key structure associated with the private key. As shown in Figure 8, the decryption time is proportional to the number of leaf nodes needed for decryption, and the level of the access tree has no impact on the decryption time. Obviously, here also the time taken to perform the operation of the proposed system is less compared to that of prior work.

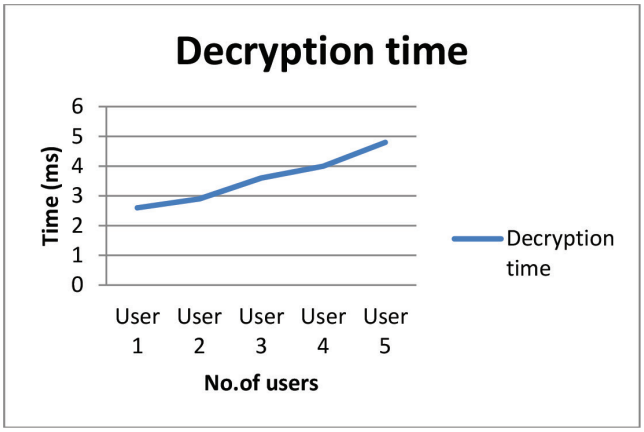

Figure 8. Decryption time.

5 Conclusion

Cloud Security is an important part of the cloud computing technologies where most of the data is being shared. The cloud users are unaware about the data where being stored and its security parameters. Prior works have been facilitated in devising the encryption and decryption cost of cloud storage systems. In this paper, we have proposed an improved and efficient file hierarchical attribute based encryption process which devises the standard of encryption and decryption process. In first step, the files are arranged in hierarchical form, so as to easy retrieval process. File Index is maintained for every uploaded file which is accessed in tree based structure. In second step, an Improved Ciphertext Policy- Attribute Based Encryption that devises the access control layer, where we have efficiently arranged encrypted files in hierarchical model with reduced storage and time cost of encryption and decryption. Experimental analysis is evaluated in terms of time taken for setup, key generation, encryption and decryption.

6. References

1. Wang S, Zhou J, Liu JK, Yu J, Chen J, Xie W. An efficient file hierarchy attribute-based encryption scheme in cloud computing. IEEE Transactions on Information Forensics and Security; 2016. Crossref. Crossref.
2. K. Priyadarsini, Selvan KT. A survey on encryption schemes for data sharing in cloud computing. International Journal of Computer Science, Information Technology, and Security. 2012 Oct; 2(5).
3. Antony N, Melvin AAR. A survey on encryption schemes in the clouds for access control. International Journal of Computer Science and Management Research. 2012 Dec; 1(5).
4. Khan AR. Access control in cloud computing environment. ARPN Journal of Engineering and Applied Sciences. 2012 May; 7(5).
5. Zhu Y, Huy H, Ahny G-J, Huangy D, Wang S. Towards temporal access control in cloud computing. INFOCOM; 2012
6. Yu S, Ren CWK, Lou W. Achieving secure, scalable, and fine-grained data access control in cloud computing. INFOCOM10.
7. Wang G, Liu Q, Wub J, Guo M. Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers. 2011 Jul.

8. Begum R, Kumar RN, Kishore V. Data Confidentiality Scalability and Accountability (DCSA) in cloud computing. 2012 Nov; 2(11).
9. Hota C, Sanka S. Capability-based cryptographic data access control in cloud computing. International Journal on Advanced Networking and Applications. 2011; 03(03):1152–61.
10. Suma V, Kuma KV. An efficient scheme for cloud services based on access policies. International Journal of Engineering Research & Technology. 2012 Oct; 1(8).
11. Kandukuri R, Paturi VR, Rakshit A. Cloud security issues. Proceedings of the 2009 IEEE International Conference on Services Computing; 2009 Sep. p. 517–20. Crossref.
12. Sterritt R. Autonomic computing. Innovations in Systems and Software Engineering. 2005; 1(1):79–88.
13. Buyya R, Yeo CS, Venugopal S, Broberg J, Brandic I. Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems. 2008 Jun; 25(6): 599–616.
14. Vaquero LM, Rodero-Merino L, Caceres J, Lindner M. A break in the clouds: Towards a cloud definition. ACM SIGCOMM Computer Communication Review. 2009 Jan; 39(1):50–5. Crossref.
15. Salesforce.com, Inc. Force.com platform [Internet]. [cited 2009 Dec]. Available from: http://www.salesforce.com/tw/.