Improved Attribute Proxy Re-encryption Scheme

Wang Zhan[1], Yuling Chen[1,4(✉)], Wei Ren[3], Xiaojun Ren[4], and Yujun Liu[2]

[1] State Key Laboratory of Public Big Data, College of Computer Science and Technology, Guizhou University, Guiyang 550025, China
ylchen3@gzu.edu.cn
[2] School of Computer Science, China University of Geosciences, Wuhan 430074, China
[3] Blockchain Laboratory of Agricultural Vegetables, Weifang University of Science and Technology, Shouguang 262700, China
[4] Technical Center of Beijing Customs, Beijing, China

Abstract. With the development of cloud computing, storing and sharing medical data in the cloud is envisioned as a promising method for supporting a large scale of users. However, if the sensitive information contained in the medical data (e.g., prescribed drug) is leaked, the user privacy will be damaged. In this paper, we propose an attribute-based Proxy Re-encryption scheme based on data split. Security of data storage and the matching between doctors and patients are tackled by CP-ABE and Proxy Re-encryption. To protect patients' sensitive data, improve the security of files, and reduce computing overhead, we adopt medical file splitting technology and attribute key tree to update keys. The data is stored by two cloud servers. Even if either server is attacked, the clear text cannot be recovered without key information. Besides, mixed encryption is used to improve the operation efficiency.

Keywords: Data sharing · Cloud storage · Attribution-based encryption · Proxy Re-encryption · Data split

1 Introduction

With the rapid development of Cloud computing technology, more and more users choose to store and share their files in the Cloud, which is more convenient and efficient than traditional storage. At present, most of Cloud storage services are provided by third-party to help users manage information and store files. Now medical data sharing has become more and more popular. However, if the patient's medical records are leaked by the platform administrator or stolen by hackers, it will inevitably bring the risk of privacy disclosure to doctors and patients. Therefore, in order to ensure the safe storage of patients' medical data and facilitate the communication between doctors and patients, reducing the pressure on the server is the research hotspot now [1, 2].

In different environments, many scholars have proposed CP-ABE schemes with different functions and backgrounds. Blaze [3] and others first proposed the concept of

J. Xiong et al. (Eds.): MobiMedia 2021, LNICST 394, pp. 343–353, 2021.
https://doi.org/10.1007/978-3-030-89814-4_25

Attribute encryption, which provides a new form of data sharing. Sahai [4–6] and others put propose an identity-based encryption scheme.

In 2012, Seo [7] and others proposed the Proxy Re-encryption scheme (ABPRE), which combines the original attribute encryption scheme (ABE) with the Proxy Re-encryption, so that the sender can decrypt the data by satisfying the attributes when the sender is offline.

The contributions of our work in this paper are shown as follow:

1) A new scheme based attribute Proxy Re-encryption scheme is proposed. After file split, the data is divided into Body Data and Sensitive data, and the Body Data is encrypted symmetrically and stored in the Data Cloud. Symmetric encryption is used to encrypt the Body Data to update the key and reduce the calculation cost. The CP-ABE and Proxy Re-encryption are used to be responsible for the operation and matching, and then the cloud server can not get the complete file.
2) Owing to patients' different requirements, it is necessary to update the key of medical records. But the key update of Asymmetric encryption is inefficient. The Body Data is encrypted symmetrically, the key and ciphertext are stored separately, which is more efficient. This paper designs an attribute key tree for attribute encryption to achieve the purpose of key update, we adopt the attribute key generation tree to generate the keys of doctors and other roles.
3) Using the characteristics of Proxy Re-encryption and secondary encryption, users can expand the matching requirements (e.g. patients can not solve the disease through traditional interrogation). According to the patients' new requirements, a new shared structure is introduced by patients, and remote person consultation can be realized.

2 Related Work

Ibraimi et al. [8] proposed an access control scheme with Proxy Re-encryption technology, but the cost of calculating Proxy Re-encryption key in the user revocation process remains to be solved. Liu [9] proposed that in the cloud computing environment, the cloud service is composed of multiple servers, and the user's data is often stored in multiple servers, and the user's encryption operation may not be executed by all servers. Liang K et al. [10] proposed the attribute-based Proxy Re-encryption (CP-ABPRE), which extends the traditional Proxy Re-encryption (PRE) by allowing semi trusted agents to convert the ciphertext under the access policy into the text with the same Plain Text under another access policy (i.e., attribute-based re-encryption). Tiwari et al. [11] provide a flexible encrypted access control mechanism for data security access. A Proxy Re-encryption scheme based on ciphertext policy attribute is proposed. Niu et al. [12] proposed an improved Proxy Re-encryption sharing scheme, which improved the Proxy Re-encryption to store the medical records in the cloud server. Zhang et al. [13] proposed a sharing scheme of cloud storage combined with block chain based on attribute agent re-encryption. Luo et al. [14] used the cross domain multi authorization center to share the key, and used the key separation technology to realize the data privacy protection. However, the above scheme can not be used to update the key, and there is a great security risk. This scheme proposes a storage scheme of medical records based on attribute Proxy

Re-encryption. The medical records are divided into two parts, the Body Data and the Sensitive Data. Compared with the existing scheme, it reduces the operation efficiency and improves the security.

3 Preliminaries

3.1 System Role

The scheme mainly include six roles: Patient, Hospital, Doctor, Key Distribution Center, Data Cloud and Proxy Cloud, Process of our split attribute Proxy Re-encryption, which is shown in Fig. 1.

1) Patient: Responsible for submitting attributes to the key distribution center and managing key pairs.
2) Hospital (HS): The hospital is responsible for generating patient medical records for split and encryption.
3) Doctor: Responsible for receiving medical records and submitting their own attributes to the key distribution center.
4) Key Distribution Center (KDC): Responsible for accepting the attributes of doctors and patients, and generating all keys.
5) Data Cloud (DC): A third-party cloud service provider with huge storage space and the ability to store large-scale data.
6) Proxy Cloud (PC): A third-party cloud service provider with powerful computing capabilities, mainly responsible for Proxy computing.

4 Protocol and Algorithm Design

4.1 Scheme Design

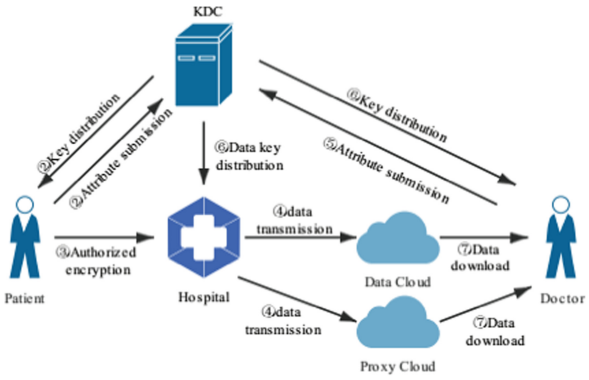

Fig. 1. Technological process of our split attribute Proxy Re-encryption.

The variable symbols used in the specific scheme, which is shown in Table 1.

Table 1. Symbolic variable

Symbol	Meaning
U	Attribute set
PK, SK	Public key and private key
GP	Public parameters
MSK	Master key
k_*	Data key
(M^*, ρ^*)	Shared structure
$rk_{A\to B}$	Proxy Re-encryption key
CT, m	Ciphertext, Plain Text
I	Key information

4.2 Algorithm Component

There are eight algorithms in this scheme:

$$setup1^k, U \rightarrow GP, MSK_*, PK \tag{1}$$

System Initialization Algorithm: Input security parameters and attribute sets of 1^k and U respectively, output GP as public parameters, MSK and PK as system master key and system public key according to security parameters and attribute set.

$$\begin{aligned} &Keygen_1(PK, MSK, S_A) \rightarrow PK_A, SK_A \\ &Keygen_2(PK, MSK, S_B) \rightarrow PK_{\boldsymbol{B}}, SK_B \\ &Keygen_3(MSK_*, Tree) \rightarrow k_* \end{aligned} \tag{2}$$

Key Generation Algorithm: Input system public key PK, system master key MSK and user submitted attribute S_A,S_B. Output public key and private key PK_A,PK_B and SK_A, SK_B. Master key MSK and attribute key $Tree$ to generate Data key k_*.

$$DataSegm \rightarrow m_1, m_2 \tag{3}$$

Data Split: The Clear text m is divided into two parts, including Sensitive data m_1, body data m_2.

$$ReKeygen_1(GP, SK_A, \left(M^{'}, \rho^{'}\right), PK_B) \rightarrow rk_{A\rightarrow B} \tag{4}$$

Re-encryption Key Generation: Input public parameter GP, user private key SK_A. And the shared structure $\left(M^{'}, \rho^{'}\right)$ and public key PK_B, Output Generate Proxy Re-encryption key $rk_{A\to B}$.

$$\begin{aligned} &Encrypt_1(m_1, k_*) \to CT_A^{'} \\ &Encrypt_2(GP, m_2, k_*, I, M, \rho, PK) \to CT_A \end{aligned} \tag{5}$$

Information Encryption Algorithm 1: Input Body data m_1, Data key k_*. Output the ciphertext $CT_A^{'}$.

Information Encryption Algorithm 2: Input Public parameter GP, Sensitive data m_2, shared structure (m, ρ),the public key PK and data key k_* And key information I. Output ciphertext CT_A.

$$ReEncrypt(rk_{A\to B}, CT_A, \left(M^{'}, \rho^{'}\right), PK_B, SK_A) \to CT_B \tag{6}$$

Ciphertext Re-encryption Algorithm: Judge whether Bob is the system contract user, if Bob is the system contract user, Input ciphertext CT_A. Bob public key PK_B, the patient's private key SK_A. Share structure $\left(M^{'}, \rho^{'}\right)$, and Proxy Re-encryption key $rk_{A\to B}$ generate re-encrypted ciphertext CT_B.

$$ReDecryptCT_B, SK_B \to m_2, k_* \tag{7}$$

Re-encryption and Decryption Algorithm: The system checks whether Bob conforms the shared structure$\left(M^{'}, \rho^{'}\right)$, If Bob $\left(M^{'}, \rho^{'}\right)$ consistent with Bob, Bob can use his private key SK_B to Decrypt CT_B get sensitive data m_2 and data key k_*.

$$\begin{aligned} &Decrypt_1(CT_A, PK, SK_A) \to m_2 \\ &Decrypt_2\left(CT_A^{'}, k_*, GP\right) \to m_1 \end{aligned} \tag{8}$$

Ciphertext Decryption algorithm: If the submitted attribute conforms to the shared structure, Bob obtains the private key SK through the key Distribution Center (KDC). According to the PK, SK get the sensitive datam_2. Bob asks cloud for $CT_A^{'}$, using the data key k_*, And decrypt$CT_A^{'}$, get the datam_1.

4.3 Proposed Procedures

1) **Initialization stage**
Run algorithm 1, randomly select groups G and G_T and generators g, $g_1 \in G$,α, $a \in Z^*_p$ taking Bilinear mapping $e : G \times G \to G_T$, Generate the system public parameter GP and Hash function to represent the identity of the Hash function ID, the role of the basic attributes of the Hash function H_1. The Hash function of doctor's basic information H_2. Hash function of message H_3.

$$\begin{aligned} &GP = \left(p, g, G, G_T, e, g_1, g^a, H_1, H_2, H_3, ID, e(g, g)^{\alpha}\right) \\ &ID : \{0, 1\}^* \to G, H_1, H_2, H_3 : G_T \to Z_p^* \\ &PK = (e(g, g)^a, g, g_1, g^a),\ MSK = \left(g^{\alpha}, a\right) \end{aligned} \tag{9}$$

Patients applied to the KDC and randomly selected $t \in Z^*_P$. Run algorithm 2 to generate the unique public and private key pairs of patients (doctors) (Fig. 2).

$$SK = (K = g^{at} g^{\alpha}, L = g^{t}, K_x = ID(x)^{t}, t \in S_A) \tag{10}$$

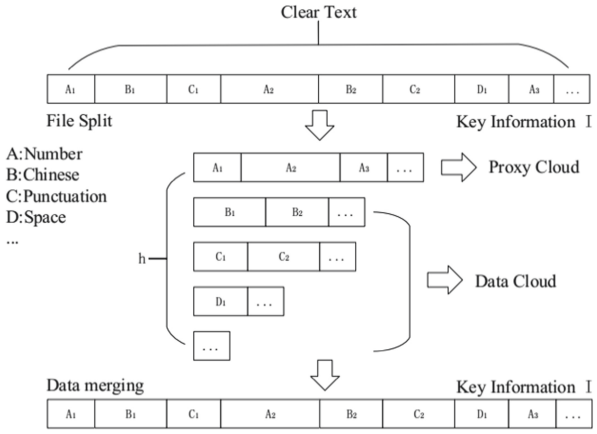

Fig. 2. Data split of our scheme.

2) **Encryption stage**
Run the algorithm 5 to encrypt and store it in the cloud. The data key is generated by the Key Distribution Center (KDC).The structure of the attribute based key tree is shown in the Figs. 3 and 4.

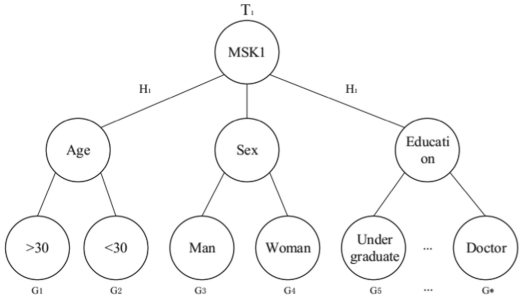

Fig. 3. The attribute based key tree of our scheme. (part 1)

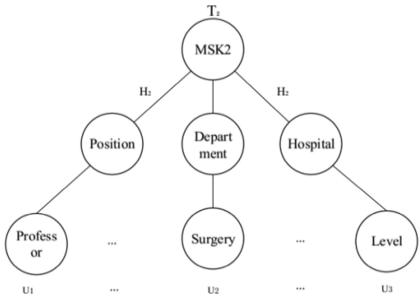

Fig. 4. The attribute based key tree of our scheme. (part 2)

At the same time, each node represents a different attribute group, and the leaf node represents each user's attribute. Each user has a unique identifier *ID*. For example, a 40 years old man with degree is a professor of surgery in a specialized hospital. After using the key to encrypt the segmented data segment, each user has his own k k_*. The key k is:

$$k = H_1\left(K_{G_1} \| K_{G_3} \| K_{G_5}\right) \| H_2\left(K_{U_1} \| K_{U_2} \| K_{U_3}\right) \tag{11}$$

When the user no longer authorized the doctor access rights or authorized to the next doctor, the key was changed, and the previously authorized user could not access the file normally. Proxy Cloud receives data m_1. Run the algorithm 5 to encrypt the data, in which the access shared structure set by the patient's M, ρ . Only when the attribute access applied by the doctor meets the shared structure can ciphertext be obtained CT_B. The encrypted ciphertext can be expressed as follows:

$$\begin{aligned} CT_A &= ((m, \rho), A_1, A_2, A_3, (B_1, C_1) \cdots (B_i, C_i)) \\ A_1 &= (m_2 + I + k_*) \cdot e(g, g)^{\alpha \cdot s}, A_2 = g^s, A_3 = g_1^s \\ B_1 &= \left(g^{\alpha}\right)^{\gamma_1} \cdot ID(\rho(1))^{-r_1}, \cdots, \\ B_i &= \left(g^{\alpha}\right)^{\gamma_i} \cdot ID(\rho(1))^{-r_i} \\ C_1 &= g^{r_1}, \cdots, C_i = g^{r_i} \end{aligned} \tag{12}$$

Where M is the matrix of $L \times N$, ρ is the mapping of M rows to attributes, and $\{\rho(i)|1 \leq i \leq l\}$ denotes the attributes in the access structure (m, ρ). S is the secret to be shared. $s, y_2, y_3 \cdots y_n \in Z^*{}_P$. For $i = 1$ to l, set $\gamma_1 = vM_i$,M_i is the vector corresponding to row i of matrix M, $v = (s, y_2, y_3 \cdots y_n)$,$r_1 \cdots r_i \in Z^*{}_P$ [15].

3) **Proxy key generation and Re-encryption stage**

At this time, it is necessary to determine whether the doctor attribute conform the new attribute set by the patients' $\left(M', \rho'\right)$. If the access structure is conformed, run algorithm 2.2 to generate the doctor's Public key and Private key. Among them, the Proxy Cloud randomly selects $\theta \in Z^*{}_P$. so that all doctors who conform the shared structure can obtain the same data key and realize consultation. The hospital (HS)

calculation of $rk_{A\to B}$.

$$\begin{aligned} &rk_{A\to B} = (rk_1, rk_2, rk_3, rk_4, R_X) \\ &rk_1 = K^{H_3(\delta)} g_1^{\theta} = g^{\alpha} g^{at} g_1^{\theta}, \\ &rk_2 = g^{\theta}, rk_3 = L^{H_3(\delta)}, \\ &rk_4 = C'_{(m',\rho')}, R_X = K_X^{H_3(\delta)} \end{aligned} \tag{13}$$

Judge whether the doctor's attribute conform the shared structure $\left(M^{'}, \rho^{'}\right)$. Random selection $\delta \in G_T$.

$$\begin{aligned} &CT_B = (\left(m^I, \rho^I\right), A'_1, A'_2, A'_3, \left(B'_1, C'_1\right) \\ &\cdots \left(B_i^{'}, C_i^{'}\right), A_4, rk_4) \\ &A_1^{'} = \delta \cdot e(g,g)^{\alpha \cdot s'}, A_2^{'} = g^{s'}, A_3^{'} = g_1^{s'} \\ &B_1^{'} = \left(g^{\alpha}\right)^{\gamma_1^{'}} \cdot ID(\rho(1))^{-r_1^{'}}, \cdots, \\ &B_i = \left(g^{\alpha}\right)^{\gamma_i^{'}} \cdot ID(\rho(1))^{-r_i^{'}} \\ &C_1^{'} = g^{r_1^{'}}, \cdots, C_i = g^{r_i^{'}} \end{aligned} \tag{14}$$

When s conform the shared structure of $\left(M^{'}, \rho^{'}\right)$, there exists a set of constants $\{\omega_i \in Z^*{}_P\}_{i\in l}$, make $\sum_{i\in l} w_i \gamma_i = s$, Where $\{\gamma_i\}$ is the secret of S.

$$A_4 = \frac{\frac{e(A_2, rk_1)}{e(A_3, rk_2)}}{\prod_{i\in l}(e(B_i, rk_3)e(C_i, R_{p(i)}))^{\omega_i}} \tag{15}$$

4) **Decryption stage**

When the user communicates with the doctor, the doctor submits the corresponding attributes to the key distribution system and conform the shared structure, the algorithm 8 is used to decrypt the encrypted ciphertext. The solution process is as follows.

$$\begin{aligned} &\frac{\frac{A_1}{e(A_2,K)}}{(\prod_{i\in l}\left(e(B_i, L)e\left(C_i, R_{p(i)}\right)\right)^{\omega_i})} \\ &= \frac{(m_2 + I + k_*) \cdot e(g,g)^{\alpha \cdot s}}{e(g,g)^{\alpha \cdot s}} \\ &= m_2 + I + k_* \\ &D_{k_*}(CT_*) = m_1 \\ &m_2 + m_1 = m \end{aligned} \tag{16}$$

When the doctor obtains segment m_2. Data key k_*. After that, Body data m_1 and sensitive data m_2 to get clear text M. When the user is not satisfied with the doctor's

diagnosis, or the doctor can not make an accurate judgment, the solution process is as follows.

$$\delta = \frac{\frac{A_1'}{e\left(A_2', K'\right)}}{\left(\prod_{i \in l}\left(e\left(B_i', L'\right)e\left(C_i', R_{p(i)}'\right)\right)^{\omega_i'}\right)} \tag{17}$$

$$(m_2 + I + k_*) = \frac{A_1}{(A_4)^{\frac{1}{H_3(\delta)}}}$$

5 Experiments and Analysis

5.1 Performance Analysis

In this paper, we simulate our experiment in a computer with Intel i5–6500 CPU @ 3.20 Hz and 8 GB memory windows, and test the algorithm efficiency with Python.

Then, we compared the efficiency and ciphertext length of attribute with Luo's and Zhang's scheme. Suppose that S represents the number of attributes and L_G, L_{G_T}, L_{Z_P} represent the length of G, G_P, Z_P respectively. We mainly discuss four components of our proposed scheme: system public key, system master key, user private key and ciphertext length (Table 2).

Table 2. Complexity comparison

Scheme	Luo's	Zhang's	Our's
PK	L_{G_T}	$(S+3)L_G + L_{G_T}$	$L_{G_T} + 3L_G$
MSK	L_G	$L_G + L_{Z_P}$	$L_G + L_{Z_P}$
SK	$(3S+2)L_G$	$(2S+2)L_G$	$(S+2)L_G$
CT	$SL_G + L_{G_T}$	$(2S+2)L_G + L_{G_T}$	$(S+1)L_G + L_{G_T}$

E is the exponentiation time, B is the bilinear mapping operation, S is the attribute, E_k is the time encrypted (Table 3).

Table 3. Efficiency comparison

Schemes	Luo's	Zhang's	Our's
Initialization	$E(G) + E(G_T) + B$	$(S+3)E(G) + E(G_T) + B$	$2E(G) + E(G_T) + B + hH(m_1)$
Key generation	$(2+S)E(G)$	$(3+S)E(G)$	$(2+S)E(G) + (h-1)E_k$
Encryption	$(10S+4)E(G) + 5E(G_T) + 9B$	$(3S+3)E(G) + 3E(G_T) + 7B$	$\frac{(6S+)E(G)+9B}{h} + \frac{h-1}{h}E_k$
Decryption	$SB + \gamma E(G)$	$(4S+6)E(G) + 5B$	$\frac{SB+\gamma E(G)}{h} + \frac{h-1}{h}E_k$

Figure 5 shows that the number of attributes has no impact on our proposed scheme in the initialization phase, compared with Luo's scheme, our proposed scheme performs

file and data segment integrity hash verification in the initialization phase. Figure 6 shows that in the key generation stage, when the number of attributes increases, the computing time increases. In the key generation stage, the key update based on attribute will be executed, and the time consumed is different from other schemes. Figure 7 shows that the encryption time has some impacts with the number of attributes. The operation time of sensitive data after CP-ABE and Proxy Re-encryption is 1/h of the original file size, the total time of main data encryption, and the encryption efficiency of this scheme is improved by 34.24% compared. Figure 8 shows the time cost in decryption phase. And, in the decryption phase, our proposed scheme needs to decrypt the main data, and therefore it is more than others when the number of attributes is low.

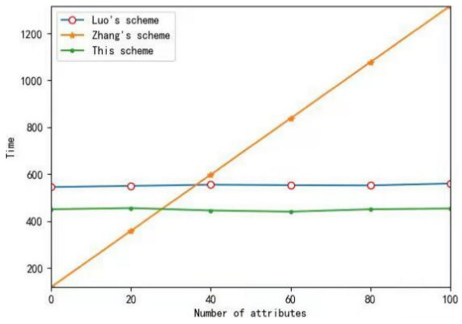

Fig. 5. Initialization stage.

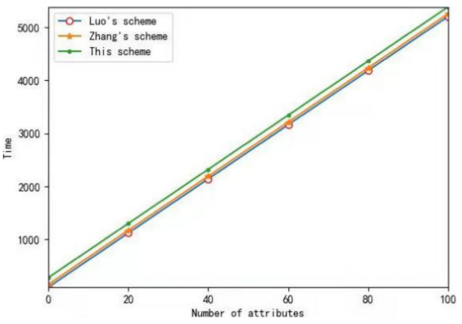

Fig. 6. Key generation stage.

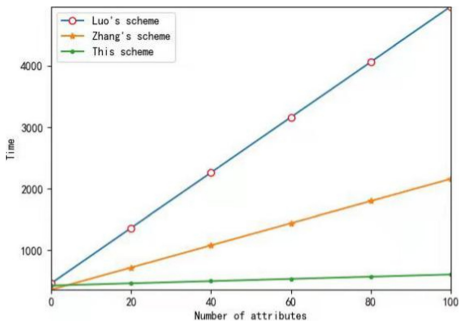

Fig. 7. Encryption stage.

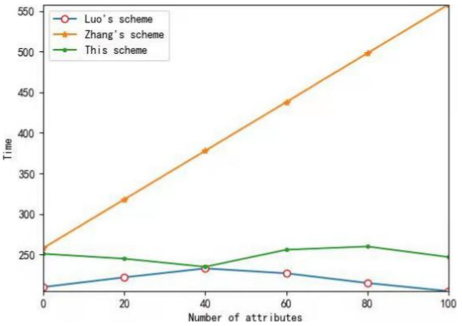

Fig. 8. Decryption stage.

6 Conclusion

This scheme divides the data into different clouds for storage, reduces the server pressure, and solves the efficiency problem to a certain extent. The attribute key tree is used to realize the access control of one key at a time. All these make it possible for the safe storage of medical data. Nowadays, the application of cloud storage is constantly updated, and various technologies are constantly implemented. But the high complexity of asymmetric encryption can not be widely used.

Acknowledgment. This work is supported by the National Quality Infrastructure project that is Key R&D Program of China under Grant 2018YFF0212106. and supported in part by the National Natural Science Foundation of China under Grant 61962009. In part by the Major Science and Technological Special Project of Guizhou Province under Grant 20183001. and in part by the Open Funding of Guizhou Provincial Key Laboratory of Public Big Data under Grant 2018BDKFJJ013.

References

1. Feng, C.-S., Qin, Z.-G., Yuan, D.: Techniques of secure storage for cloud data. Chinese J. Comput. **38**(01), 150–163 (2015)
2. Fang, B., Jia, Y., Li, A., Jiang, R.: Privacy preservation in big data:a survey. Big Data Res. **2**(01), 1–18 (2016)
3. Blaze, M., Bleumer, G., Strauss, M.: Divertible protocols and atomic proxy cryptography. In: Nyberg, K. (eds.) Advances in Cryptology — EUROCRYPT 1998. EUROCRYPT 1998. LNCS, vol. 1403, pp. 127–144. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0054122
4. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 457–473. Springer, Heidelberg (2005). https://doi.org/10.1007/11426639_27
5. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: 2007 IEEE Symposium on Security and Privacy (SP 2007), Berkeley, CA, pp. 321–334 (2007). https://doi.org/10.1109/SP.2007.11
6. Goyal, V., Pandey, O., Sahai, A., et al.: Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of the 13th ACM Conference on Computer and Communications Security, pp. 89–98 (2006)
7. Seo, H.J., Kim, H.W.: Attribute-based proxy re-encryption with a constant number of pairing operations. J. Inf. Commun. Converg. Eng. **10**(1), 53–60 (2012)
8. Ibraimi, L., Petkovic, M., Nikova, S., Hartel, P., Jonker, W.: Mediated ciphertext-policy attribute-based encryption and its application. In: Youm, H.Y., Yung, M. (eds.) WISA 2009. LNCS, vol. 5932, pp. 309–323. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-10838-9_23
9. Liu, Q., Tan, C.C., Wu, J., Wang, G.: Reliable re-encryption in unreliable clouds. In: 2011 IEEE Global Telecommunications Conference - GLOBECOM 2011, Houston, TX, USA, pp. 1–5 (2011). https://doi.org/10.1109/GLOCOM.2011.6133609
10. Liang, K., Fang, L., Susilo, W., Wong, D.S.: A ciphertext-policy attribute-based proxy re-encryption with chosen-ciphertext security. In: 2013 5th International Conference on Intelligent Networking and Collaborative Systems, Xi'an, 2013, pp. 552–559 (2013). https://doi.org/10.1109/INCoS.2013.103
11. Tiwari, D., Gangadharan, G.R.: SecCloudSharing: secure data sharing in public cloud using ciphertext-policy attribute-based Proxy Re-encryption with revocation. Int. J. Commun. Syst. **31**(5), e3494 (2018)
12. Niu, S., Liu, W., Chen, L., Du, X.: An electronic medical record data sharing scheme based on Proxy Re-encryption. Comput Eng. 1–10 (2020)
13. Zhang, X., Sun, L.: Attribute proxy re-encryption for ciphertext storage sharing scheme on blockchain. J. Syst. Simul. **32**(06), 1009–1020 (2020)
14. Luo, E., Wang, G., Chen, S., Pinial, K.: Privacy preserving friend discovery cross domain scheme using re-encryption in mobile social networks. J. Commun. **38**(10), 81–93 (2017)
15. Liu, M., Liu, S., Wang, Y., Wang, J., Li, Y.,, Cao, H.: Optimizing the decryption efficiency in LSSS matrix-based attribute-based encryption without given policy. Chinese J. Electron. **43**(6), 1065–1072 (2015)