ITAIC 2020 (ISSN 2693-2865)

Attribute-Based Keyword Search Encryption Scheme with Verifiable Ciphertext via Blockchains

ShuFen Niu
College Of Computer Science And Engineering
Northwest Normal University
Lanzhou 730070, China

LiXia Chen
College Of Computer Science And Engineering
Northwest Normal University
Lanzhou 730070, China

WenKe Liu
College Of Computer Science And Engineering
Northwest Normal University
Lanzhou 730070, China

2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC) | 978-1-7281-5244-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/ITAIC49862.2020.9338962

***Abstract*—In order to realize the sharing of data by multiple users on the blockchain, this paper proposes an attribute-based searchable encryption with verifiable ciphertext scheme via blockchain. The scheme uses the public key algorithm to encrypt the keyword, the attribute-based encryption algorithm to encrypt the symmetric key, and the symmetric key to encrypt the file. The keyword index is stored on the blockchain, and the ciphertext of the symmetric key and file are stored on the cloud server. The scheme uses searchable encryption technology to achieve secure search on the blockchain, uses the immutability of the blockchain to ensure the security of the keyword ciphertext, uses verify algorithm guarantees the integrity of the data on the cloud. When the user's attributes need to be changed or the ciphertext access structure is changed, the scheme uses proxy re-encryption technology to implement the user's attribute revocation, and the authority center is responsible for the whole attribute revocation process. The security proof shows that the scheme can achieve ciphertext security, keyword security and anti-collusion. In addition, the numerical results show that the proposed scheme is effective.**

***Index Terms*—Blockchain; Attribute-based encryption, Searchable encryption, Verifiable ciphertext**

I. INTRODUCTION

While cloud storage brings convenience to users, it also brings a series of personal data security issues. The data in the cloud storage is out of the control of the user, and there is a lack of trust between the user and the cloud service provider. The data needs to be encrypted before being stored in the cloud server.In view of the above problems, Sahai and Waters[1] proposed a fuzzy identity-based encryption scheme in 2005, and Goyal[2] extended it to attribute encryption, providing efficient and fine-grained access control while realizing data privacy [3]. In addition, how to retrieve encrypted data is another problem for ABE [4] in practical application. Song et al. [5] first proposed Searchable Encryption (SE) scheme based on stream password and symmetric Encryption. By referring to the encryption features of CP-ABE system, Li et al. [6] proposed a searchable encryption scheme based on attributes for the first time, and realized that the encrypted data could be searched by multiple parties. However, data stored in cloud servers may be tampered with, and the integrity is not guaranteed. The rise of blockchain technology [7-8] can well solve these problems.In 2019, Wu et al. [9] proposed an efficient and traceable attribute-based encryption scheme on the block chain. This scheme hides the access structure and realizes the data integrity and non-repudiation through the block chain technology.

The main contributions of this paper include three aspects:

- First, fine-grained access control is realized by using attribute-base encryption technology. The scheme USES wildcard to hide access structure, and reencryption by proxy to realize attribute revocation.
- Second, cloud server and block chain are used in the construction of the scheme. The cloud server stores data ciphertext and symmetric key ciphertext, while the block chain stores key ciphertext.In this paper, the untamperability of block chain is used to guarantee the data integrity of the ciphertext of key words, while the verification algorithm and the ciphertext of key words ensure that the ciphertext on the cloud is not tampered.
- Third, secure search on the blockchain is realized by using searchable encryption technology. Patients authorize doctors to handle electronic medical records, and doctors store the ciphertext of keywords on the blockchain.When the user accesses ciphertext, the patient generates a search trap door and hands it to the user. The user uploads the trap door to the block chain, and the node on the block chain conducts keyword ciphertext search.

II. SYSTEM AND SECURITY MODEL

A. SYSTEM MODEL

The system structure of the scheme is shown in figure 1. There are five entities involved in the system formework: Authority center, date owner, blockchain, cloud and data user.

1) AUTHORITY CENTRE: The authority center is fully trusted by the other four entities. The authority center is in charge of system initialization. Moreover, authority center take charge of issuing attribute keys for users.

2) DATE OWNER: The date owner is responsible for encrypting data and uploading it to the medical cloud and blockchain. The data owner firstly encrypts the health record by symmetric encryption algorithm and encrypts the symmetric key by public key encryption technology. At the same time, the data owner extracts keywords set from the file and establishes the keywords index. Moveover, the data owner signs the keywords index. then data owner uploads the keywords

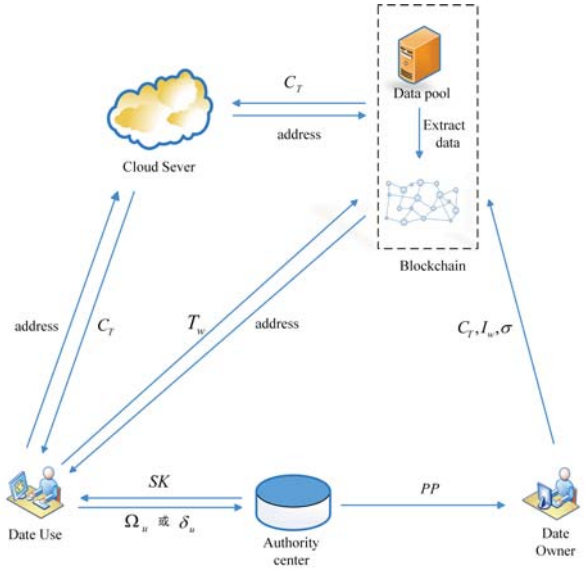

Fig. 1: System model

index, the ciphertext and signatures to the data pool of the blockchain.

3) BLOCKCHAIN: In order to get bitcoin, the consensus nodes will keep monitoring the data pool, they extract the keywords index and signatures from the data pool, and verify that the keywords index matches the signatures. Then, they perform a consensus protocol to select a bookkeeping node. The bookkeeping node submits the ciphertext to cloud and get the data address form the cloud. Moveover, the bookkeeping node submits the keywords index along with the data address in the specified format to the blockchain. Through the incentive mechanism, system selects the corresponding blockchain node as the verifier to run the search algorithm. If success, the blockchain node return the data address.

4) CLOUD: The cloud is in charge of data storage. When the bookkeeping node submits the ciphertext to cloud, these data is stored in the cloud and cloud return data address. If the data user wants to share the ciphertext, the data user upload data address to cloud, then cloud extracts ciphertext to date user by the dara address.

5) DATA USER: When the data user wants to obtain the data, the data user can access data address on the blockchain by providing a search trapdoor generated with an interested keyword and his private key. Then the date user can access file on the cloud by the date address. Finally, the date user can decrypts the ciphertext by using of his private key.

B. Consensus Mechanism

As it determines whether the new block is validated and who keeps the record, so consensus mechanism is the core technology. A blockchain network usually consists of data producers, consensus nodes, and data pool. In our scheme, date owner submit keywords index I_w, signature σ and ciphertext C_T to the data pool as data producers. Then consensus nodes verify check whether the keywords index I_w are complete by the signature σ. After verifying these data, the consensus node runs the consensus protocol and selects the bookkeeping node. The bookkeeping node is responsible for writing ciphertext C_T to the cloud and writing keywords index I_w and signature σ to the blockchain.

C. ACCESS STRUCTURE

Let Ω be the universe of possible attributes that can be expressed as $att_1, att_2, ..., att_n$, where the index n denotes the number of attributes. For each attributes att_i, denote $V_i = \{\nu_{i,1}, \nu_{i,2}, ..., \nu_{i,m}\}$ as the universe of possible attribute values, where the index m denotes the number of possible values for the attribute att_i. This paper use wildcard $*$ in access structure to denote the 'do not care' value to protect privacy.

III. OUR SCHEME

In the section, the attribute-based keyword search encryption scheme with verifiable ciphertext via blockchain is proposed and and notations used are defined in Table.1,The proposed scheme is composed of four phases: system setup, data storage and index generation, data sharing, attribute revocation.

TABLE I: Notations

Nations	Description
Ω	All of the attribute of system
Ω_u	All of the attributes of user
Ω_o	All of the attributes of access policy
δ_u	The unique identity of data user
$\overline{v_{i,j}}$	The revocation attribute set of user
k	Symmetric key of AES
C_F	Ciphertext of Symmetric key
C_T	Encrypted file set
I_w	Keywords index
$W = (w_1, ..., w_t)$	Keywords set extracted by owner
$W' = (w_1, ..., w_{t'})$	Keywords set interested by user

A. Phase 1: System setup

When a data user registers, authority center select randomly δ_u as the unique identity of data user, and the unique identity δ_u are public. Then, the authority center runs algorithm 1 to generate system parameter PP and the master key Msk, where there have a number of attribute value for each attribute att_i. By running algorithm 2, private key sk are generated for data user. The date user have an attribute set Ω_u, for each attribute att_i, date user has only a unique value $v_{i,j}$.

setup(κ, Ω): The algorithm is run by the authority center. Given a security parameter κ and attributes set Ω, the system parameter PP are public, and the master key Msk is kept in authority center itself for generating private keys of authorized users. The process is illustrated in Algorithm 1.

KeyGen(Ω_u, δ_u): The algorithm is run by the authority center. Given the attribute set Ω_u of data user and the unique identity δ_u of data user, then outputs private keys of data user. The process is illustrated in Algorithm 2.

Algorithm 1: Setup

Input: Security parameter κ, attributes set Ω.
Output: System parameters PP, master key Msk.

Select multiplicative cyclic groups G_1 and G_2, $e : G_1 \times G_1 \to G_T$, g, h is a generator of G_1;
Select hash functions: $H_1 : Z_p^* \to G_1$, $H_2 : G_T \to Z_p^*$, $H_3 : Z_p^* \times Z_p^* \to Z_p^*$, $H_3 : G_1 \times G_2 \to Z_p^*$;
Select randomly $\omega \in Z_p^*$ the unique identity of data user $\delta_u \in Z_p^*$;
Calculate $Y_1 = g^{\omega}$, $Y_2 = e(g,h)^{\omega}$;
for $att_i \in \Omega$ **do**
  for $v_{i,j} \in att_i$ **do**
    Select randomly $a_{i,j} \in Z_p^*$, $b_{i,j} \in Z_p^*$, $A_{i,j} \in G_1$;
    Calculate $A_{i,j}^{a_{i,j}}$, $A_{i,j}^{b_{i,j}}$;
$\{\omega, a_{i,j}, b_{i,j}\} \leftarrow Msk$;
$\{g, h, G_1, G_2, e, Y_1, Y_2, \delta_u, H_1, H_2, H_3, H_4, \{A_{i,j}^{a_{i,j}}\}, \{A_{i,j}^{b_{i,j}}\}\} \leftarrow PP$;
return Msk, PP;

Algorithm 2: KeyGen

Input: Attribute set Ω_u of data user, the unique identity δ_u of data user.
Output: Private key sk.

Select randomly $\lambda_i \in Z_p^*$;
Calculate $d = \omega/\delta_u$;
for $v_{i,j} \in \Omega_u$ **do**
  Calculate $D_i = h^{\omega} A_{i,j}^{a_{i,j} b_{i,j} \lambda_i}$, $P_i = g^{a_{i,j}\lambda_i}$, $R_i = g^{b_{i,j}\lambda_i}$;
$\{d, \{D_i, P_i, R_i\}\} \leftarrow sk$;
return sk;

B. Phase 2: Data storage and index generation

When a data owner encrypts a file F, he proceeds as follows:

Enc(PP, W, k, o): The algorithm is run by the data owner. Given Public parameters PP, keywords set $W = (w_1, ..., w_t)$ of the file F, the file F, symmetric key k and access structure o. Firstly, data owner encrypts extracts keywords set $W = (w_1, ..., w_t)$ and build an encrypted keywords index for the file. Secondly, data owner encrypts the data file F with a symmetric key of AES, and denotes the encrypted result as C_F. In addition, data owner defines an access policy o about the symmetric key εk and encrypts εk under o. Finally, For the keywords index I_w, data owner computes the signature of keywords index : $\sigma = \{S_1, S_2\}$.
Algorithm 3 describes the steps involved in generating index I_w, generating ciphertext C_T of the symmetric key and file and signature σ.

Algorithm 3: Date encryption

Input: Public parameters PP, file F and keywords set $W = (w_1, ..., w_t)$, symmetric key εk, access structure o.
Output: Index I_w, ciphertext C_T, signature σ.

Select randomly $r_w \in Z_p^*$, $r \in Z_p^*$, $r_\sigma \in Z_p^*$;
Calculate $B = g^{r_w}$;
for $1 < k < t$ **do**
  Calculate $I_j = e(H_1(w_k)^{r_w}, Y_1)$;
Set $\{B, \{I_j\}\} \leftarrow I_w$;
Encrypts the file F with symmetric key εk of AES, $C_F \leftarrow Enc_{\varepsilon k}(F)$;
Calculate $C_0 = k \oplus H_2(Y_2^r)$, $C_1 = g^r$, $C_2 = H_3(C_0, C_F) \times r$;
for $v_{i,j} \in \Omega_o$ **do**
  Select randomly $\gamma_i \in Z_p^*$;
  Calculate $E_{i,j} = (A_{i,j}^{b_{i,j}})^{\gamma_i}$, $L_{i,j} = (A_{i,j}^{a_{i,j}})^{r-\gamma_i}$;
$\{C_F, C_0, C_1, C_2, \{E_{i,j}, L_{i,j}\}\} \leftarrow C_T$;
Calculate $S_1 = H_4(I_w) \times r_\sigma$, $S_2 = h^{r_\sigma}$;
$\{S_1, S_2\} \leftarrow \sigma$;
returnI_w, C_T, σ;

The data owner submits keywords index I_w, ciphertext C_T and signature σ to the data pool of the blockchain. The consensus nodes extract keywords index I_w and signature σ from the data pool, calculates $H_4(I_w)$ and check whether the equations $e(Y_1, S_2)^{H_4(I_w)} = Y_2^{S_1}$. If the equation holds, the hospital is authorized by the patient, the consensus nodes broadcasts a verification confirmation message.

Then, they perform a consensus protocol to select a bookkeeping node. The bookkeeping node submits the ciphertext C_T and the unique identity δ_u to the cloud and get the data address $addrese$ form the cloud. Finally, the bookkeeping node write the keywords index I_w, signature σ and the data address in the specified format as figure 2 to the blockchain.

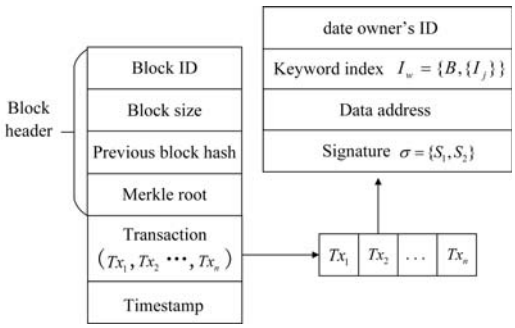

Fig. 2: Data structure

C. Phase 3: Data sharing

When data user wants to access the file F, the data user is required to use the private key to generate the searching

trapdoor by run the algorithm 4. Through the incentive mechanism, system selects the corresponding blockchain node as the verifier to run algorithm 5.

Trapdoor(sk, W'): The algorithm is run by the data user. Given the private key sk of data user and keywords set $W' = (w_1, ..., w_{t'})$, then computes trapdoor as $T = (\prod H_1(w_j))^d$. The process is illustrated in Algorithm 4.

Algorithm 4: Trapdoor generation

Input: Private key sk, keywords set $W' = (w_1, ..., w_{t'})$.
Output: Search token T.

```
for 1 < k < t' do
    Calculate T = (∏ H1(wk))^d;
return T;
```

Search(I_w, T, δ_u): When the blockchain node access the trapdoor submitted by data user and the data user has a unique identity δ_u, the blockchain node Checks whether $e(B,T)^{\delta_u} = \prod I_j$. If the equation holds, the blockchain node can obtains the data address $addrese$ by accessing the block which stores this index. Then the blockchain node returns it to the date user. Algorithm 5 describes the steps involved in the search.

Algorithm 5: Search

Input: Index I_w, Search token T.
Output: success or $\perp$.

```
for 1 < k < t do
    if e(B,T)^{δu} = ∏ Ik then
        Output success;
    else
        Output ⊥ ;
```

Verify(C_T): When the data user wants to access file F, he get ciphertext $C_T = \{C_F, C_0, C_1, C_2, \{E_{i,j}, L_{i,j}\}\}$ on the cloud by the data address. To verify whether data in the cloud has been tampered with, the data user calculates $H_3(C_0, C_F)$, checks whether $e(g^{c_2}, g) = e(g^{H_3(C_0,C_F)}, C_1)$. If the equation holds, data in the cloud has not been tampered with. The process is illustrated in Algorithm 6.

Algorithm 6: Verify

Input: ciphertext C_T, Private key sk.
Output: Success or $\perp$.

```
Calculate H3(C0, CF);
if e(g^{c2}, g) = e(g^{H3(C0,CF)}, C1) then
    Output Success;
else
    Output ⊥ ;
```

Decrypt(C_T, sk): Algorithm 7 gives the steps of user decryption. Firstly, the data user calculates $\tau = \frac{\prod e(D_i, C_1)}{\prod e(P_i, E_{i,j}) \prod e(R_i, L_{i,j})}$ and the symmetric key $\varepsilon k = C_0 \oplus H_2(\tau)$. Second, data user decrypt the ciphertext with symmetric key εk. Finally, data user calculates file $F = Dec_{\varepsilon k}(C_F)$.

Algorithm 7: Decrypt

Input: ciphertext C_T, Private key sk.
Output: file F.

```
Calculate τ = ∏ e(Di, C1) / (∏ e(Pi, Ei,j) ∏ e(Ri, Li,j));
Calculate symmetric key εk = C0 ⊕ H2(τ);
Calculate F = Dec_{εk}(F);
return F;
```

IV. CORRECTNESS

In this section, we show the correctness of our scheme. The correctness of consensus mechanism:

$$\begin{aligned} e(Y_1, S_2)^{H_4(I_w)} &= e(g^{\omega}, h^{r_\sigma})^{H_4(I_w)} \\ &= Y_2^{S_1} \end{aligned}$$

The correctness of search:

$$\begin{aligned} e(B,T)^{\delta_u} &= e(g^{r_w}, (\prod H_1(w_j))^d)^{\delta_u} \\ &= e(g^{r_w}, (\prod H_1(w_j))^{\omega/\delta_u})^{\delta_u} \\ &= \prod I_j \end{aligned}$$

The correctness of verify:

$$\begin{aligned} e(g^{c_2}, g) &= e(g^{H_3(C_0,C_F)\times r}, g) \\ &= e(g^{H_3(C_0,C_F)}, C_1) \end{aligned}$$

The correctness of decryption:

$$\begin{aligned} \tau &= \frac{\prod e(D_i, C_1)}{\prod e(P_i, E_{i,j}) \prod e(R_i, L_{i,j})} \\ &= \frac{\prod e(h^{\omega} A_{i,j}^{a_{i,j} b_{i,j} \lambda_i}, g^r)}{\prod e(g^{a_{i,j}\lambda_i}, (A_{i,j}^{b_{i,j}})^{\gamma_i}) \prod e(g^{b_{i,j}\lambda_i}, (A_{i,j}^{a_{i,j}})^{r-\gamma_i})} \\ &= \frac{e(h^{\omega}, g^r) \prod e(A_{i,j}^{a_{i,j} b_{i,j} lambda_i}, g)^r}{\prod e(A_{i,j}^{a_{i,j} b_{i,j} lambda_i}, g)^r} \\ &= e(h^{\omega}, g^r) \\ &= Y_2^r \end{aligned}$$

$$\begin{aligned} k &= C \oplus H_2(\tau) \\ &= k \oplus H_2(Y_2^r) \oplus H_2(Y_2^r) \\ &= k \end{aligned}$$

$$\begin{aligned} F &= Dec_k(C_F) = Dec_k(Enc_k(F)) \\ &= F \end{aligned}$$

V. SECURITY ANALYSIS

The scheme satisfies keyword privacy security and ciphertext privacy security base on the decision linear problem and bilinear diffie-Hellman problem. However, due to space constraints, it will not be repeated.

VI. Conclusion

The scheme realizes fine-grained access control on blockchain, and Supports attribute revocation, secure search and privacy protection. The authority center implements attribute revocation and updates only those attribute that the user needs to revoke. The scheme realizes keyword search on blockchain. The data integrity of key ciphertext is guaranteed by the unforgeability of blockchain. the data integrity of ciphertext on the cloud is guaranteed by verify algorithm. In addition, the scheme is proved to be ciphertext security and keyword security. This scheme can be applied to other scenarios, such as electronic medical records, Internet of vehicles.

Acknowledgment

This work was supported by the National Natural Science Foundation of China under Grants No. 61562077, No. 61662071, No. 61662069 and No. 61772022.

References

[1] A. Sahai, B. Waters, "Fuzzy identity-based encryption," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, Berlin, Heidelberg, Nov.2005, pp.457-473.

[2] V. Goyal, O. Pandey, A. Sahai, 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*, Acm, Nov.2006, pp.89-98.

[3] Y. Wang, F. Fang, "Effective CPABE with Hidden Access Policy," *Journal of Computer Research and Development*, vol.56, pp.2151-2159., 2019.

[4] J. Liu , X. Huang , J. Liu, "Secure sharing of Personal Health Records in cloud computing: Ciphertext-Policy Attribute-Based Signcryption," *Future generation computer systems*, vol.52, pp.67-76, 2015.

[5] D. X. Song, D. Wagner, A. Perrig, "Practical techniques for searches on encrypted data," in *Proceeding 2000 IEEE Symposium on Security and Privacy*, IEEE, 2000, pp.44-55.

[6] J. G. Li, Y. R. Shi, Y. C. Zhang, "Searchable ciphertext policy attribute-based encryption with revocation in cloud storage," *International Journal of Communication Systems*, vol.30, pp.e2942, 2017.[1]

[7] X. Yang, T. Li, R. Liu, et al, "Blockchain-Based Secure and Searchable EHR Sharing Scheme," in *2019 4th International Conference on Mechanical, Control and Computer Engineering (ICMCCE)*, Hohhot, China: IEEE, Oct.2019.

[8] A. Zhang, X. Lin, "Towards Secure and Privacy-Preserving Data Sharing in e-Health Systems via Consortium Blockchain," *Journal of Medical Systems*, vol.42, no.8, pp.140-158, 2019.

[9] A. X. Wu, Y. H. Zhang, X. K. Zheng, et al, "Efficient and privacy-preserving traceable attribute-based encryption in blockchain," *Annals of Telecommunications*, vol.20, no.8, pp.1-11, 2019.

[10] X. Wang, A. Zhang, X. Xie, et al, "Secure-aware and privacy-preserving electronic health record searching in cloud environment," *International journal of communication systems*, vol.32, no.8, pp.e3925.1-e3925.11, 2019.