IMPLEMENTATION OF VERIFIABLE SEARCHABLE SYMMETRIC ENCRYPTION FOR DYNAMIC CLOUD DATA

[1]T. Anil Karuna Kumar **[2]T. Suresh**

[1] Associate Professor, Dept. of Master of Computer Applications, Narayana Engineering College, Gudur, AP, India.

[2] PG Scholar, Dept. of Master of Computer Applications, Narayana Engineering College, Gudur, AP, India.

Abstract – Verifiable Searchable Symmetric Encryption, as an important cloud security technique, allows users to retrieve the encrypted data from the cloud through keywords and verify the validity of the returned results. Dynamic update for cloud data is one of the most common and fundamental requirements for data owners in such schemes. The overhead of verification may become a significant burden due to the sheer amount of cloud data. Therefore, how to achieve keyword search over dynamic encrypted cloud data with efficient verification is a critical unsolved problem. To address this problem, we explore achieving keyword search over dynamic encrypted cloud data with symmetric-key based verification and propose a practical scheme in this paper. In order to support the efficient verification of dynamic data, we design a novel Accumulative Authentication Tag (AAT) based on the symmetric-key cryptography to generate an authentication tag for each keyword. Benefiting from the accumulation property of our designed AAT, the authentication tag can be conveniently updated when dynamic operations on cloud data occur.

Keywords – Searchable Symmetric Encryption, Encrypted Cloud Data, Verification, Data Dynamic.

I. INTRODUCTION

Searchable Symmetric Encryption (SSE) is a practical way for users to securely retrieve the interested ciphertextsfrom the encrypted cloud data through keywords. It hasbecome a hot research topic in cloud computing security andnumerous SSE schemes have been proposed. Nonetheless,most of them only consider realizing keyword search overstatic encrypted cloud data. In practice, the data stored on thecloud server might often need to be updated (added, deletedor modified) by data owners. Therefore, it is necessary todesign SSE schemes supporting dynamic update for clouddata. Kamara et al. [1] proposed a SSE scheme supportingdata dynamic update. This scheme designs a search table byextending the inverted index to realize the sublinear search,and adopts a search array and a deletion array with otherfree storage spaces to achieve data dynamics. Guo et al. [2]proposed a dynamic SSE scheme, in which an inverted indexis used to record the locations of keywords. The update tableand the update list make the scheme support data dynamics. Inaddition, some other dynamic keyword search schemes [3–5],which adopt tree-based (i.e. KBB tree, KRB tree and B+ tree)index structure, have also been proposed.

In this paper, we explore how to achieve keyword searchover dynamic encrypted cloud data with symmetric-key basedverification. The contributions of this paper can be summarizedas follows:

- In order to support the efficient verification of dynamic data, we design a novel symmetric-key based Accumulative Authentication Tag (AAT) to generate an authentication tag for each keyword. Benefiting from the accumulation property of our designed AAT, the authentication tag can be conveniently updated when dynamic operations on cloud data occur. The proposed AAT is collision resistant, that is, it is computational-difficult for any adversary to find different messages with the same tag. It also can resist the replay attack to prevent the cloud server from returning the old data that actually has been updated.

- In order to realize efficient data update, we design a new secure index composed by a search table ST and a verification list VL. ST is based on the orthogonal list and VL is a singly linked list. For each keyword, we construct a linked list with the same length aiming at hiding the frequency of each keyword. When performing modification operations, the cloud server can fleetly find the index nodes related to the modified files. When some files need to be added or deleted, the secure index can be conveniently enlarged or reduced. Owing to the connectivity and flexibility of ST, the update efficiency can be significantly improved.
- Based on the above technique and structure, we design the first keyword search scheme over dynamic encrypted cloud data with symmetric-key based verification. We give the security analysis of the proposed scheme and conduct the performance comparison with other work in terms of the search token generation efficiency, verification efficiency and update efficiency. The results show that the proposed scheme is secure and efficient.

II. BACKGROUND WORK

In recent years, cloud computing has been applied tosecurely perform various tasks, such as healthcare monitoring, deep packet inspection and key updates. Cloudstorage is universally viewed as one of the most importantservices of cloud computing. Although cloud storage providesgreat benefit to users, it brings some new security challenges.Firstly, users may worry about whether their data is intentlystored in the cloud because the cloud data is out of their physicalcontrol. In order to solve this problem, some cloud storageauditing schemes are proposed to check the integrityof cloud data. In addition, users usually need to encrypt thedata for keeping the privacy before outsource them to thecloud. It makes performing keyword search over encryptedcloud data become a new challenge. In order to address thisissue, searchable encryption is proposed, which allows usersto selectively retrieve cipher documents stored in the cloud bykeyword-based search. Compared with searchable public key encryption searchable symmetric encryption draws more attention owing to its high efficiency.

Dynamic SSE. In order to support data dynamic update,some dynamic SSE schemes have been proposed.Kamara et al. [1] proposed a dynamic SSE schemeby extending the inverted index approach. This scheme canachieve sublinear search and CKA2-security. Subsequently,they proposed another dynamic SSE scheme [4] based onkeyword red-black tree index structure. This scheme supportsparallel keyword search as well as parallel addition anddeletion of files. Naveed et al. presented a dynamic SSEscheme via blind storage. Blind storage allows a data owner tostore files on a cloud server in such a way that the cloud serverdoes not learn the number of files. Xia et al. [3] proposed adynamic keyword search scheme over encrypted cloud databased on the tree-based index structure, which can supportmulti-keyword rank. Guo et al. [2] proposed a dynamic SSEscheme based on the inverted index. It enables the data user tosearch several phrases in a query request. Also, their proposedscheme supports the sorting of the search results.

Verifiable SSE. In order to prevent the cloud server fromreturning the invalid search results, Chai et al. firstlyproposed the verifiable keyword search scheme over encryptedcloud data. In order to make the scheme with the "verifiablesearchability", the cloud server is required to provide a proofalong with the returned results. Kurosawa et al. haveshown how to construct a universally composable (UC)-secureverifiable SSE scheme. Jiang et al. proposed a verifiablemulti-keyword ranked search scheme over encrypted clouddata. A special data structure named QSet is constructed toachieve efficient keyword search in this scheme. In order tosupport dynamic data update for verifiable SSE schemes, Sunet al. [10] proposed a verifiable dynamic conjunctive keywordssearch scheme based on the bilinear-map accumulator and theaccumulation tree. Zhu et al. [9] introduced a verifiable anddynamic fuzzy keyword search scheme based on the invertedindex. Liu et al. [7] presented a verifiable dynamic keywordsearch scheme supporting the search results rank. This schemeand scheme [9] both leverage RSA accumulator to realize theresults verification and the data dynamics. The verificationtechniques used in above verifiable and dynamic schemes areall based on asymmetric-key cryptography, which involvestime-consuming operations. As a result, the verification efficiencyis very low in these schemes.

III. PROPOSED WORK

System Model

As shown in Fig.1, the system model consists of three entities: data owner, data user and cloud server.

Data owner: He encrypts his plain files and constructs a secure index with private keys. He uploads the ciphertexts and the secure index to the cloud server. When the data owner wants to update files, he generates the update tokens locally and sends them to the cloud server.

Data user: He is authorized by the data owner who shares the private keys with him. When he wants to search the files containing the interested keywords, he sends the search requests to the cloud server. After the data user receives the search results from the cloud server, he can verify the validity of the results.

Cloud server: It stores the ciphertexts and the secure index from the data owner. Upon receiving the search requests from the data user, it performs search operation over the secure index, and returns the search results. In addition, upon receiving the update information from the data owner, it updates the secure index and the related ciphertexts.

In this model, the data owner and the data user are assumed to be always trusted. That is, the data owner honestly encrypts files and builds a secure index. The data user honestly generates the search request for the queried keyword. The cloud server is regarded as an untrusted entity. It is allowed to learn which encrypted files contain the queried keyword by performing the search operation. However, it might try to learn more valuable information from the encrypted files, the secure index, and the search trapdoors. For example, it might try to find which files contain two queried keywords and which keywords have changed in the modified file. Besides, the cloud server may return invalid or non-updated search results to the data user for saving computation cost or other reasons.

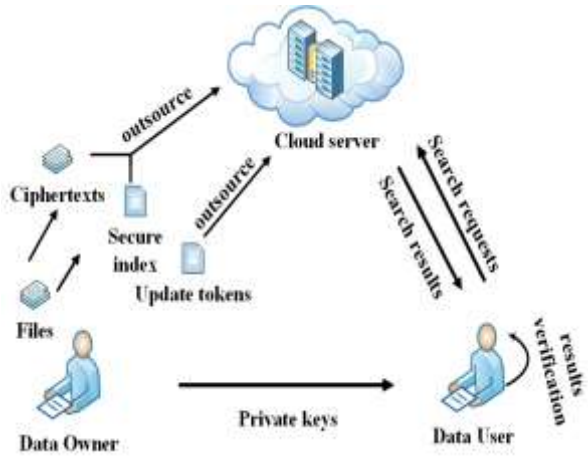

Fig. 1: System Overview

A verifiable and dynamic SSE scheme includes eight polynomial-time algorithms i.e. Setup, IndexBuild, GenToken, Search, Verify, Dec, UpToken and Update. These algorithms are defined as follows.
IndexBuild(K,F,W) is the probabilistic index building algorithm run by the data owner. It takes the private key set *K*, the file set F and the keyword set *W* as input, and outputs a secure index *I* and a ciphertext collection *C*.

GenToken (K,w) is the (possibly probabilistic)
trapdoor generation algorithm run by the data user. It takes the private key set *K* and the queried keyword w as input, and outputs the trapdoor T_w.
Search (T_w, *I, C)* is the deterministic search algorithm run by the cloud server. It takes the trapdoor T_w, the secure index *I* and the ciphertext set *C* as input, and outputs a ciphertext set *C(w)* and an authentication tag AATS.
Verify (*K*, T_w, *C(w), AATS)* is the deterministic verification algorithm run by the data user. It takes the private key set *K*, the trapdoor T_w, the set *C(w)* and the authentication tag AATS as input, and outputs \accept" or \reject".
Dec (K, C(w)) is the deterministic decryption algorithm run by the data user. It takes the private key set *K* and the set *C(w)* as input, and outputs a plaintext set *F(w).*
UpToken (K, F, (F_0)*)* is the (possibly probabilistic) update tokens generation algorithm run by the data owner. When modifying a file, it takes as input the original file *F*, the new file *F'*and the private key set *K*, and outputs the modify token *T*.
Update (T, I, C) is the deterministic update algorithm run by the cloud server. It takes as input the update token, the secure index *I*, and the ciphertext collection *C*. It outputs a new secure index *I'*, and a new ciphertext collection *C'*.

IV. RESULTS AND DISCUSSION

Index construction efficiency. To evaluate the efficiency of our proposed schemes, we conduct the experiments for building ST and building VL. Fig.2 shows the time cost of building the secure index when the number of files is set to 10000 and the number of keywords varies from 1000 to 10000.

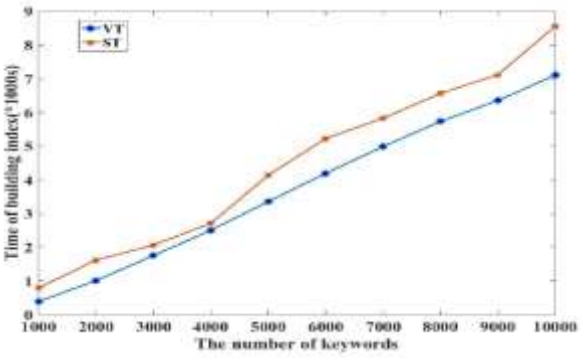

Fig. 2: Secure index construction time cost

Update token generation efficiency. Fig.3 shows that the generation time for update tokens, i.e., modify token, add token and delete token, is almost linear with the number of keywords.

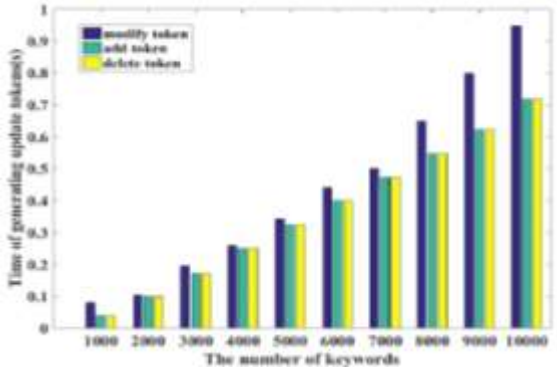

Fig. 3: Modify token generation time cost

Search efficiency. The cloud server respectively performs search operations in ST and VL. In Fig.4, we vary the number of files from 1000 to 10000 and set the number of keywords to 10000.

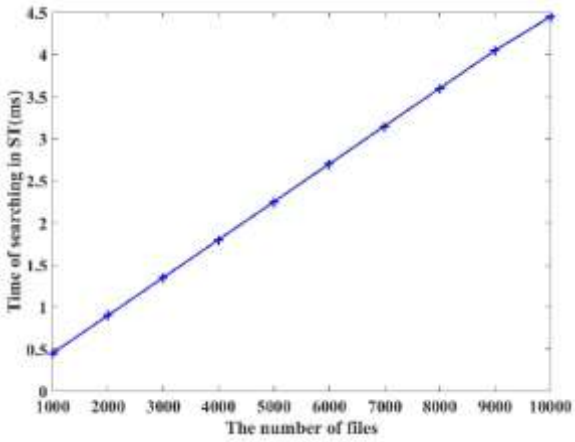

Fig. 4: Search time cost in ST

V. CONCLUSION

In this paper, we explore realizing keyword search over dynamic encrypted cloud data with symmetric-key based verification. In order to support the efficient verification of dynamic data, we design a novel Accumulative Authentication Tag (AAT) based on symmetric-key cryptography to generate an accumulative authentication tag for each keyword. Moreover, a new secure index based on the orthogonal list and the single linked list is designed to improve the updated efficiency. The security analysis and the performance evaluation show that the proposed scheme is secure and efficient.

REFERENCES

1. S. Kamara, C. Papamanthou and T. Roeder, “Dynamic searchable symmetric encryption,” presented at ACM Conference on Computer and Communications Security, pp. 965-976, 2012.
2. C. Guo, X. Chen, Y. M. Jie, Z. J. Fu, M. C. Li and B. Feng, “Dynamic Multi-phrase Ranked Search over Encrypted Data with Symmetric Searchable Encryption,” in IEEE Transations on Services Computing, vol. 99, No. 1939, pp. 1-1, 2017.
3. Z. H. Xia, X. H.Wang, X. M. Sun and Q.Wang, “A Secure and Dynamic Multi-Keyword Ranked Search Scheme over Encrypted Cloud Data,” in IEEE Transactions on Parallel and Distributed Systems, vol. 27, No. 2, pp. 340-352, 2016.
4. S. Kamara and C. Papamanthou, “Parallel and Dynamic Searchable Symmetric Encryption,” presented at the International Conference on Financial Cryptography and Data Security, pp. 258-274, 2013.
5. J. B. Yan, Y. Q. Zhang and X. F. Liu, “Secure multikey word search supporting dynamic update and ranked retrieval,” in China Communication, vol. 13, No. 10, pp. 209-221, 2016.
6. K. Kurosawa and Y. Ohtaki, “How to Update Documents Verifiably in Searchable Symmetric Encryption,” presented at International Conference on Cryptology and Network Security, pp. 309-328, 2013.
7. Q. Liu, X. H. Nie, X. H. Liu, T. Peng and J. Wu, “Verifiable Ranked Search over dynamic encrypted data in cloud computing,” presented at the IEEE/ACM International Symposium on Quality of Service, pp. 1-6, 2017.

8. X. H. Nie, Q. Liu, X. H. Liu, T. Peng and Y. P. Lin, "Dynamic Verifiable Search Over Encrypted Data in Untrusted Clouds," presented at the International Conference Algorithm and Architectures for Parallel Processing, pp. 557-571, 2016.
9. X. Y. Zhu, Q. Liu and G. J. Wang, "A Novel Verifiable and Dynamic Fuzzy Keyword Search Scheme over Encrypted Data in Cloud Computing," presented at the IEEE Trustcom/BigDataSE/ISPA, pp. 845-851, 2017.
10. W. H. Sun, X. F. Liu, W. J. Lou. Y. T. Hou and H. Li, "Catch you if you lie to me: Efficient verifiable conjunctive keyword search over large dynamic encrypted cloud data," presented at the IEEE Conference on Computer Communications (INFOCOM), pp. 2110-2118, 2015.

Author's Profile:

T. Anil Karuna Kumar has received his PG degree in *Master of Computer Applications* from *R.V.R & J.C College of Engineering*, affiliated to *Acharya Nagarjuna University, Guntur.* At present he is working as an *Associate Professor* in Narayana Engineering College,Gudur, AndhraPradesh, India.

T. Suresh has received his B.Sc degree in *Computer Science* from Vidyalaya Degree College, Gudur affiliated to *VikramaSimhapuri University, Nellore* in 2017 and pursuing PG degree in *Mater of Computer Applications* (MCA) from Narayana Engineering College, Guduraffiliated to *JNTU, Ananthapur,* AndhraPradesh, India.