International Journal on Applications in Engineering and Technology
Volume 9 : Issue 2 : April 2023, pp 45 - 49 www.aetsjournal.com ISSN (Online) : 2455 - 0523

ENABLING VERIFIABLE AND DYNAMIC RANKED SEARCH OVER OUTSOURCED DATA

P.NATARAJAN , MOVIGA .B

Abstract **-** The project **"ENABLING VERIFIABLE AND DYNAMIC RANKED SEARCH OVER OUTSOURCED DATA" i**s developed to Currently, Cloud computing as a promising computing paradigm is increasingly utilized as potential hosts for users' massive dataset. Since the cloud service provider (CSP) is outside the users' trusted domain, existing research suggests encrypting sensitive data before outsourcing and adopting Searchable Symmetric Encryption (SSE) to facilitate keyword-based searches over the ciphertexts. However, it remains a challenging task to design an effective SSE scheme that simultaneously supports sublinear search time, efficient update and verification, and on-demand information retrieval. To address this, we propose a Verifiable Dynamic Encryption with Ranked Search (VDERS) scheme that allows a user to perform top-K searches on a dynamic document collection and verify the correctness of the search results in a secure and efficient way. Specifically, we first provide a basic construction, VDERS0, where aranked inverted index and a verifiable matrix are constructed to enable verifiable document insertion in top-K searches. Then, an advanced construction, VDERS⋆, is devised to further support document deletion with a reduced communication cost. Extensive experiments on real datasets demonstrate the efficiency and effectiveness of our VDERS scheme.

I. INTRODUCTION

As a promising computing paradigm, cloud computing has drawn great attention from both research and industry communities. Because of the benefits of low costs, flexibility, and scalability, it has become a prevalent trend for users to outsource their massive datasets to clouds and delegate a cloud service provider (CSP) to manage data storage and offer query services. Due to security and privacy concerns, existing research suggests encrypting data before outsourcing. However, data encryption makes keyword-based searches over ciphertexts a challenging problem. This is even harder for efficient top-K searches in a dynamic and malicious cloud environment.

Let us consider the following scenario. Alice outsources archived emails to the cloud, where each email is indexed by the sender's name and ranked in descending order of the receipt date. For example, for a set of emails indexed by keyword Bob, the email received on April 2 has a higher rank than the email received on April 1. To keep keyword and document contents secret, Alice uploads them in the encrypted forms to the cloud. There could be hundreds of documents matching a specific keyword, and the consumed costs will be extensive if all the matched documents are returned to and decrypted by the user. Therefore, Alice may want to perform a top-K search to retrieve the most recent emails. Moreover, Alice may want to store only the emails received in the last three months for monetary saving. For example, when entering May, Alice will delete all emails received before February.

In the above application scenario, the adopted encryption scheme should meet the following requirements: (1) Ranked search. The user is allowed to perform a top-K search to retrieve the best-matched documents. (2) Dynamic. The user is able to update (add and delete) documents stored in the cloud. (3) Verifiability. The malicious CSP may delete encrypted documents not commonly used to save memory space, or it may forge the search results to deceive the user. Even if the CSP is honest, a virus or worm may tamper with encrypted documents. Therefore, the user should have the ability to verify the correctness of the search results. (4) Efficiency. The user can efficiently perform searches, updates, and verifications on a set of encrypted documents.

P.Natarajan , Assistant Professor , Department of Computer Applications , Erode Sengunthar Engineering College (Autonomous), Perundurai , Erode.
(Email : palanisamynatarajan50@gmail.com)
Moviga.B, PG Scholar , Department of Computer Applications, Erode Sengunthar Engineering College (Autonomous), Perundurai , Erode. (Email : moviga69@gmail.com).

Although Searchable Symmetric Encryption (SSE) allows a user to retrieve desired documents in a privacy-preserving way, existing SSE schemes only partially address the above requirements. To simultaneously satisfy all these properties, this paper proposes a Verifiable Dynamic Encryption with Ranked Search (VDERS) scheme that allows the user to perform updates and top-K searches on cipher texts in a verifiable and efficient way. Our main idea is to construct a verifiable matrix to record the ranking information and encode it with RSA accumulator [3]. Furthermore, a ranked inverted index is built from a collection of documents to facilitate efficient top-K searches and updates. Specifically, we first provide a basic construction, denoted by VDERS0, which enables verifiable document insertion operations. Then, we provide an advanced construction, denoted by VDERS⋆, which not only can support efficient deletion operations, but also can reduce communication costs without outsourcing the verifiable matrix. Our main contributions are summarized as follows:

- We propose a VDERS scheme to achieve dynamic and ranked searches in a cloud environment in an efficient and verifiable way.
- Two constructions are provided to achieve efficient top-K searches with support for verifiable updates.
- We theoretically analyze the security and performance of our scheme and conduct extensive experiments on real datasets to validate its effectiveness.

Paper organization. We introduce related work in Section 2 and provide the preliminaries in Section 3. After the overview of this work in Section 4, we provide our basic and advanced VDERS constructions in Section 5 and Section 6,

respectively. We evaluate the proposed scheme in Section 7. Finally, we conclude the paper in Section 8.

II. OBJECT DETECTION- AN OVERVIEW

1) SYSTEM MODEL

The system consists of three different parties: the CSP, the data owner, and the data user, as illustrated in Fig. 1. The CSP maintains cloud platforms that pool hard and soft resources to provide data storage and query services.

The data owner first creates ciphertexts**c** for a document collection **d**. Given keywords **w** extracted from **d**, she then builds a secure index I for fast searches, and generates a local evidence Ψ and remote auxiliary information Φ for verifiable searches. After uploading (**c**, I, Φ) to the cloud, she can perform updates on ciphertexts with an update token T∗and retrieve documents on demand with a search token TW in a verifiable way. On receiving the search results and a search proof (RW, Π) from the CSP, the data owner recovers document contents after verifying the correctness of the search results. The data owner can also delegate the search/update/verification ability to authorized data users. In this paper, we do not differentiate between the data owner and the data user, and refer to them as users.

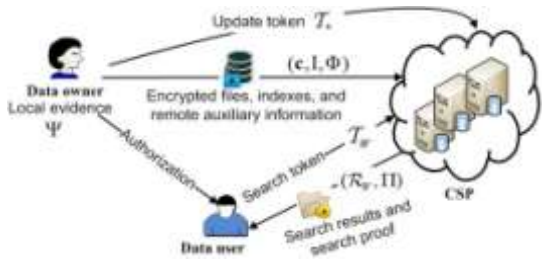

Figure 1. System model. Communication channelare secured with security protocols such as SSL/SSH.

2) ADVERSARY MODEL

We assume that the users are fully trusted. The CSP is the potential attacker and is assumed to be honest but curious [1]. That is, the CSP would correctly execute the prespecified protocol, but still attempt to learn extra information about the stored data and the received message.

As defined in [5], access pattern refers to the outcome of search results, i.e., which documents have been returned; search pattern refers to whether two searches have been performed for the same keyword. As a tradeoff between security and efficiency, existing SSE schemes resort to the weakened security guarantee for efficiency concerns.

That is, they will reveal the access pattern and the search pattern but nothing else during the search process. Like existing SSE schemes, such

information is also available to the CSP in our scheme. Furthermore, in a top-K search, only K highest ranked documents will be returned. Therefore, our scheme will leak information about document ranks besides access pattern and search pattern. It is worth noticing that the leakage of ranking information is inevitable in top-K searches. For example, the adversary first issues a top-1 search and then a top-2 search for keyword W, thereby it can know that the document returned in the first round has the

highest rank and the new document returned in the second round is ranked second. If the adversary issues top-1, 2,topcontinuously, then the rank of each document will be exposed by comparing search results.

Our scheme mainly aims to preserve the following privacy properties:

Confidentiality. The CSP knows nothing about the document/keyword contents expect search pattern, access pattern, and document ranks.

Verifiability. The CSP cannot forge a search result or falsify the outsourced ciphertexts.

III. LITERATURE SURVEY

Content Based Image Retrieval is most dominant process image query. Past researches are mainly focused on improving the efficiency of CBIR, with little attention paid to privacy and security issues related to CBIR. This paper reviews existing techniques for maintain privacy in CBIR and suggest possible research directions. Index Terms: Content Based Image Retrieval, AES, encryption, decryption.

1) SYSTEM ANALYSIS

SSE has been widely researched since it was first proposed by Song et al. [4]. As a seminal work in SSE, Carmela et al. [5] provided a rigorous security definition and constructed two schemes, SSE-1 and SSE-2, based on an inverted index. Compared to SSE-1, SSE-2 is more secure, and it has been proven to be secure against adaptive chosen keyword attacks (CKA2). SSE-1 is secure against nonadaptive chosen-keyword attacks (CKA1), but yields an optimal (sublinear) search time O(r), where r is the number of documents that contain the query keyword. However, neither SSE-1 nor SSE-2 has properties of dynamism, verifiability, and ranked search.

IV. EXISTING SYSTEM

Dynamic SSE (DSSE) allows a user to update the encrypted outsourced data in an efficient and secure way. Kamara et al. [6] constructed a DSSE scheme based on an extended inverted index. Their subsequent work [7] extended it to a parallel search setting by using a red-black tree. Cash et al. [8] constructed a DSSE scheme optimized for super large datasets, but their scheme supports only efficient document insertion.

Naveed et al. [9] put forward a DSSE scheme in which a server worked as a blind storage to decrease leakage at the cost of multiple rounds of interaction. The above schemes are proven to be CKA2-secure, but leak keyword information about the newly added documents. With this leakage, the attackers can reveal the content of a past query by injecting new documents in a dataset [10]. To mitigate such an attack, Stefano et al. [11] proposed the first DSSE scheme with forward privacy, but their scheme suffers from inefficiency. Since then, forward privacy has been the main motivation of recent DSSE schemes [12], [13]. Verifiable SSE (VSSE) allows a user to verify the correctness of search results.

Kurosawa et al. [14] constructed a Universally Composable (UC)-secure VSSE scheme, in which a user can elect any malicious server's cheating behavior. While UC-security is stronger than CKA2-security, their construction requires a linear search time. Soleimani an et al. [15] presented a public VSSE scheme, which delegates a third party to accomplish the verification, but fails to support dynamic operations. The subsequent work of

Kurosawa et.al [16] extends VSSE to a dynamic environment. Their scheme employs RSA accumulator [3] to generate constant size digests/proofs, but the verification cost on the client side grows linearly with the total number of documents. To enable verifiable conjunctive keyword search over dynamic encrypted data, Sun et al. [17] exploited the bilinear-map accumulator technique to construct an accumulation tree.

Jiang et al. [18] proposed a VDSSE scheme that also utilized an accumulator tree to verify results of Boolean queries. The accumulator tree structure is more efficient than RSA accumulator in verification, but consumes more computation time for updating tree structure. Zhu et al. [19] proposed a generic VSSE scheme in a multi-user setting, where the verifiable design can provide result verification for any SSE schemes and support data updates. However, the above VSSE schemes return all search results and therefore, may be unsuitable for an environment where a lot of documents match a user's query but the user is only interested in best-match documents.

DISADVANTAGES

1) The system was not implemented Verifiable Dynamic Encryption with Ranked Search (VDERS) scheme.
2) The system is less security due to lack of Searchable Symmetric Encryption (SSE).

V. PROPOSED SYSTEM

This system proposes a Verifiable Dynamic Encryption with Ranked Search (VDERS) scheme that allows the user to perform updates and top-K searches on ciphertexts in a verifiable and efficient way. Our main idea is to construct a verifiable matrix to record the ranking information and encode it with RSA accumulator [3]. Furthermore, a ranked inverted index is built from a collection of documents to facilitate efficient top-K searches and updates. Specifically, we first provide a basic construction, denoted by VDERS0, which enables verifiable document insertion operations. Then, we provide an advanced construction, denoted by VDERS⋆, which not only can support efficient deletion operations, but also can reduce communication costs without outsourcing the verifiable matrix. Our main contributions are summarized as follows:

• We propose a VDERS scheme to achieve dynamic and ranked searches in a cloud environment in an efficient and verifiable way.

• Two constructions are provided to achieve efficient top-K searches with support for verifiable updates.

• We theoretically analyze the security and performance of our scheme and conduct extensive experiments on real datasets to validate its effectiveness.

ADVANTAGES

Confidentiality. The CSP knows nothing about the document/keyword contents expect search pattern, access pattern, and document ranks.

Verifiability. The CSP cannot forge a search result or falsify the outsourced ciphertexts.

VI. SYSTEM IMPLEMENTATION

1)CLOUD SERVER

In this module, the admin has to login by using valid user name and password. After login successful he can do some operations such Login, View All Users, View All Documents, View Top 'K' Keywords, View Keywords and Links, View Time Delay of Files, View User Transactions, View File Rank Results, View Time Delay Comparison Results.

2)DATA USER

In this module, there are n numbers of users are present. User should register with group option before doing some operations. After registration successful he has to wait for admin to authorize him and after admin authorized him. He can login by using authorized user name and password. Login successful he will do some operations like Register and Login, View Profile, Search Cloud Data, View Document Search Comparison, View Keyword and Fetched Files, View Same Data Files.

3)DATA OWNER

In this module, there are n numbers of users are present. Owner should register with group option before doing some operations. After registration successful he has to wait for admin to authorize him and after admin authorized him. He can login by using authorized user name and password. Login successful he will do some operations like Register and Login, View Profile, Upload Document, View and Edit or Delete Document.

VII. CONCLUSION

In this paper, we design a VDERS scheme to simultaneously support sub linear search time, efficient update and verification, and on-demand information retrieval in cloud computing environment. Experiment results demonstrate that our scheme is effective for verifying the correctness of top-K searches on a dynamic document collection. As part of our future work, we will try to design a forward secure VDERS scheme, where the update phase does not leak keyword information about a newly added document.

VIII. REFERENCES

[1] G. S. Pho, J.-J. Chinwag. -C. Yua, K.-K. R. Choo, and M. S. Mohamad, "Searchable symmetric encryption: designs and challenges," ACM Computing Surveys (CSUR), 2017.

[2] Q. Liu, X. Nia, X. Liu, T. Peng, and Jawun, "Verifiable ranked search over dynamic encrypted data in cloud computing," in Proc. Of Ios, 2017.

[3] J. Camelish and A. Pysanky, "Dynamic accumulators and application to efficient revocation of anonymous credentials," in Proc. of CRYPTO, 2002.

[4] D. X. Song, D. Wagner, and A. Perrigo, "Practical techniques for searches on encrypted data," in Proc. of IEEE S&P, 2000.

[5] R. Carmela, J. Gray, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: improved definitions and efficient constructions," in Proc. of the ACM CCS, 2006.

[6] S. Kamara, C. Hapaxanthous, and T. Roeder, "Dynamic searchable symmetric encryption," in Proc. of ACM CCS, 2012.

[7] S. Kamara and C. Hapaxanthous, "Parallel and dynamic searchable symmetric encryption," in Proc. of FC, 2013.

[8] D. Cash, J. Jaeger, S. Jacki, C. Juta, H. Krawczyk, M. Rous, and M. Steiner, "Dynamic searchable encryption in very-large databases: data structures and implementation," in Proc. of NDSS, 2014.

[9] M. Naveed, M. Prabhakaran, and C. A. Gunter, "Dynamic searchable encryption via blind storage," in Proc. of S&P, 2014.

[10] Y. Zhang, J. Katz, and C. Hapaxanthous, "All your queries are belong to us: The power of file-injection attacks on searchable encryption," in Proc. of USENIX Security Symposium, 2016.

[11] E. Stefano, C. Hapaxanthous, and E. Shi, "Practical dynamic searchable encryption with small leakage," in Proc. of NDSS, 2014.

[12] R. Bost, _oxo: Forward secure searchable encryption, in Proc. Of CCS, 2016.

[13] X. Song, C. Dong, D. Yuan, Q. Xu and M. Zhao, "Forward private searchable symmetric encryption with optimized I/O efficiency," in IEEE Transactions on Dependable and Secure Computing, 2018.

[14] K. Kurosawa and Y. Ohtaki, "Uc-secure searchable symmetric encryption," in Proc. of FC, 2012.

[15] Soleimani an, Azam, and S. Khazei, "Publicly verifiable searchable symmetric encryption based on efficient cryptographic components," in Designs, Codes and Cryptography, 2019.

[16] K. Kurosawa and Y. Ohtaki, "How to update documents verifiably in searchable symmetric encryption," in Proc. of CNS, 2013.

[17] W. Sun, X. Liu, W. 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," in Proc. of IEEE INFOCOM, 2015.

[18] S. Jiang, X. Zhu, L. Guo, and J. Liu, "Publicly verifiable Boolean query over outsourced encrypted data," in IEEE Transactions on Cloud Computing, 2017.

[19] J. Zhu, Q. Li, C. Wang, X. Yuan, Q. Wang and K. Ren, "Enabling generic, verifiable, and secure data search in cloud services," in IEEE Transactions on Parallel and Distributed Systems, 2018.

[20] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," IEEE Transactions on Parallel and Distributed Systems, 2014.

[21] W. K. Wong, D. W.-l. Cheung, B. Kao, and N. Mamalis, "Secure in computation on encrypted databases," in Proc, of ACM SIGMOD, 2009.

[22] W. Sun, B. Wang, N. Cao, M. Li, W. Lou, Y. T. Hou, and H. Li, "Verifiable privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking," IEEE Transactions on Parallel and Distributed Systems, 2014.

[23] C. Chen, X. Zhu, P. Shen, J. Hu, S. Guo, Z. Tari, and A. Y. Zumaya , "An efficient privacy-preserving ranked keyword search method," IEEE Transactions on Parallel and Distributed Systems, 2016.

[24] W. Zhang, Y. Lin, S. Xiao, J. Wu, and S. Zhou, "Privacy preserving ranked multi-keyword search for multiple data owners in cloud computing," IEEE Transactions on Computers, 2016.

[25] Z. Fu, K. Ren, J. Shu, X. Sun, and F. Hua, "Enabling personalized search over encrypted outsourced data with efficiency improvement," IEEE transactions on parallel and distributed systems, 2016.