International Journal of Contemporary Research in Computer Science and Technology (IJCRCST) e-ISSN: 2395-5325
Volume 4, Issue 3 (March '2018)

PHARSE SEARCH TECHNIQUE BASED ON BLOOM FILTERS FOR ENCRYPTED CLOUD STORAGE

Vinoth.V,
M.E.,(CSE),
Department of Computer Science and Engineering,
Mahendra Engineering College Autonomous,
Mahendrapuri,Namakkal Dist-637501.

Abstract: Cloud computing has generated much interest in the research community in recent years for its many advantages, but has also raise security and privacy concerns. The storage and access of confidential documents have been identified as one of the central problems in the area. In particular, many researchers investigated solutions to search over encrypted documents stored on remote cloud servers. While many schemes have been proposed to perform conjunctive keyword search, less attention has been noted on more specialized searching techniques. In this paper, we present a phrase search technique based on Bloom filters that is significantly faster than existing solutions, with similar or better storage and communication cost. Our technique uses a series of n-gram filters to support the functionality. The scheme exhibits a trade-off between storage and false positive rate, and is adaptable to defend against inclusion-relation attacks. A design approach based on an application's target false positive rate is also described.

Keywords: Cloud Computing,Security, Keyword search, Encryption, Decryption

I. INTRODUCTION

AS organizations and individuals adopt cloud technologies, many have become aware of the serious concerns regarding security and privacy of accessing personal and confidential information over the Internet. In particular, the recent and continuing data breaches highlight the need for more secure cloud storage systems. While it is generally agreed that encryption is necessary, cloud providers often perform the encryption and maintain the private keys instead of the data owners. That is, the cloud can read any data it desired, providing no privacy to its users. The storage of private keys and encrypted data by the cloud provider is also problematic in case of data breach. Hence, researchers have actively been exploring solutions for secure storage on private and public clouds where private keys remain in the hands of data owners.

Boneh et al proposed one of the earliest works on keyword searching. Their scheme uses public key encryption to allow keywords to be searchable without revealing data content. Waters et al. investigated the problem for searching over encrypted audit logs. Many of the early works focused on single keyword searches. Recently, researchers have proposed solutions on conjunctive keyword search, which involves multiple keywords Other interesting problems, such as the ranking of search results and searching with keywords that might contain errors termed fuzzy keyword search, have also been considered.

The ability to search for phrases was also recently investigated. Some have examined the security of the proposed solutions and, where flaws were found, solutions were proposed In this paper, we present a phrase search scheme which achieves a much faster response time than existing solutions. The scheme is also scalable, where documents can

II. RELATED WORK

We'll describe our keyword search framework using two parties: The data owner and an untrusted cloudserver. Our algorithms can easily be adapted to the scenario of an organization wishing to setup a cloud server for its employees by implementing a proxy server in place of the data owner and having the employees/users authenticate to the proxy server. A standard keyword search protocol is shown in figure 1. During setup, the data owner generates the required encryption keys for hashing and encryption operations. Then, all documents in the database are parsed for keywords. Bloom filters tied to hashed keywords and programs are attached. The documents are then symmetrically encrypted and uploaded to the cloud server. To add files to the database, the data owner parses the files as in setup and uploads them with Bloom filters attached to the cloud server. To remove a file from the data, the data owner simply sends the request to the cloud server, who removes the file along with the attached Bloom filters. To perform a search, the data owner computes and sends a trapdoor encryption of the queried keywords to the cloud to initiate a protocol

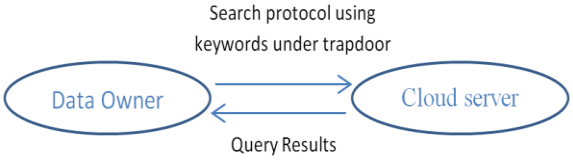

Figure 1. Communication framework for keyword search over encrypted Data

to search for the requested keywords in the corpus. Finally, the cloud responds to the data owner with the identifiers to the requested documents. Our framework differs from some of the earlier works, where keywords generally consist of meta-data rather than content of the files and where a trusted key scow authority is used due to the use of Identity based

encryption. When compared to recent works, our setup is equivalent to that of where an organization wishes to outsource computing resources to a cloud storage provider and enable search for its employees, and similar to where the aim is to return properly ranked files. Most other recent works related to search over encrypted data have considered similar models such as, where the client acts as both data owner and user. Note that, depending on the application, the encrypted documents may or may not require retrieval once the query is resolved. Should retrieval be required, further privacy issues may arise. These issues are considered in oblivious storage and private information retrieval schemes Our discussions will mainly restrict to the protocol leading to the query resolution. Direct retrieval is assumed where appropriate to better compare against existing solutions for phrase search.

2.1 Bloom filters

Researchers have proposed the use of Bloom filters for conjunctive keyword search to reduce storage cost and provide security in the form of false positives. Bloom filters are space-efficient probabilistic data structure used to test whether an element is a member of a set. A Bloom filter contains m bits, where k hash functions, Hi(x), are used to map elements to the m-bits in the filter. The Bloom filter is initially set to all zeros. To add and element, a, to the filter, we compute Hi(a) for i = 1 to k, and set the corresponding positions in the filter to 1. For example, for

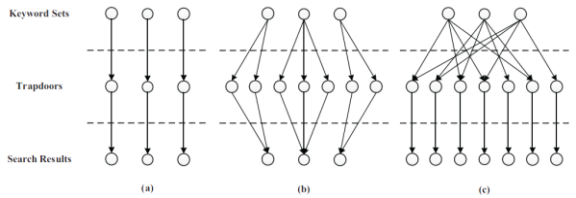

Figure. 2. Relationship between keyword set, trapdoor and search result k = 2 and m = 7, to add 'Bell' to the filter, we compute H1(Bell) = 2 and H2(Bell) = 5. Setting the position 2 and 5, the Bloom filter becomes 0; 1; 0; 0; 1; 0; 0. To test for membership of an element, b, in a sample Bloom filter, we compute Hi(b) for i = 1 to k, the element is determined to be a member if all corresponding positions of the sample Bloom filter is set to 1. For example, 'Bell' would be a member of the Bloom filter, 0; 1; 1; 0; 0; 1; 1. While Bloom filters have no false-negatives, it can falsely identify an element as member of a set. Given k hash functions, n items inserted and m bits used in the filter, the probability of false positives is approximately p = (1 □ e□kn=m)k and minimum false positive rate is achieved when k = m n ln2

2.2 Inclusion-Relation attacks

In described an inclusion-relation (IR) attack, where two sets of query results, a and b, with a being a subset of b, would imply the queried keywords that led to b includes keywords that led to a. A cloud server with some knowledge on the statistical properties of the search terms and has access to sets of trapdoors and their associated search results can potentially discover some of the keywords. Our technique can be adapted to defend against such attacks by allowing a set of keywords to map to many possible queries (trapdoors) and the inclusion of false positives in search results. Also, since different sets of keywords can have the same bits in the Bloom filter being set to 1, different keyword sets can lead to the same query. Figure 2 shows different mappings of keywords to trapdoors and search results for various conjunctive keyword search schemes. In section 4.4, we described our type C design, which was suggested to provide the best defense against IR attacks.

III. PROBLEM STATEMENT

3.1 EXITING MODEL

In this project, we present a phrase search scheme which achieves a much faster response time than existing solutions. The scheme is also scalable, where documents can easily be removed and added to the corpus. We also describe modifications to the scheme to lower storage cost at a small cost in response time and to defend against cloud providers with statistical knowledge on stored data.
Although phrase searches are processed independently using our technique, they are typically a specialized function in a keyword search scheme, where the primary function is to provide conjunctive keyword searches.

3.1.1 Limitation of existing system

- The serious concerns regarding security and privacy of accessing personal and confidential information over the Internet.
- Huge cost in terms of data usability. For example, the existing techniques on keyword-based information retrieval, which are widely used on the plaintext data, cannot be directly applied on the encrypted data. Downloading all the data from the cloud and decrypt locally is obviously impractical.
- Existing System methods not practical due to their high computational overhead for both the cloud sever and user.

3.2 PROPOSED SYSTEM

In this project, we presented a phrase search scheme based on Bloom filter that is significantly faster than existing approaches, requiring only a single round of communication and Bloom filter verifications. The solution addresses the high computational cost noted in by reformulating phrase search as n-gram verification rather than a location search or a sequential chain verification. Unlike our schemes consider only the existence of a phrase, omitting any information of its location. Unlike our schemes do not require sequential verification, is parallelizable and has a practical storage requirement.

3.2. Advantages of proposed system

According to our experiment, it also achieves a lower storage cost than all existing solutions except.
We design a searchable encryption scheme that supports both the accurate multi-keyword ranked search and flexible dynamic operation on document collection.

IV. METHODOLOGY

PHRASE SEARCH SCHEME BASED ON BLOOM FILTERS

In a keyword search scheme, Bloom filters can be used to test whether a keyword is associated with a document. Many existing phrase search schemes use a keyword-to-document index and a location/chain index to map keywords to documents and match phrases. We describe an alternative approach using Bloom filters to support this functionality with an emphasis on response time. Our scheme can be summarized as the use of multiple n-gram Bloom filters, Bn Di , to provide conjunctive keyword search and phrase search.

V. PERFORMANCE EVALUATION

our scheme requires two Bloom filters per document for the purpose of phrase search, one for storing pairs and another for triples. Note that the conjunctive keyword Bloom filters are not required for the purpose of phrase search. The two sets of filters require N(b2+b3) bits of storage on the cloud server, where b2 is the size of a pairs Bloom filter, b3 is the size of a triples Bloom filter and N is the number of documents in the corpus. The data owner needs only to store cryptography keys. Since all existing schemes include conjunctive keyword search capability, we have included the 1-gram filter, b1, in our comparison table 3 and 4. The communication cost of our scheme is similar to [13]. The proposed protocol requires only two messages to be sent, one containing the set bit locations of the query Bloom filter for pairs or triples and the other containing the matched results.

VI. CONCLUSION

This project, we presented a phrase search scheme based on Bloom filter that is significantly faster than existing approaches, requiring only a single round of communication and Bloom filter verifications. The solution addresses the high computational cost noted in reformulating phrase search as n-gram verification rather than a location search or a sequential chain verification. Our schemes consider only the existence of a phrase, omitting any information of its location. Unlike, our schemes do not require sequential verification, is parallelizable and has a practical storage requirement. Our approach is also the first to effectively allow phrase search to run independently without first performing a conjunctive keyword search to identify candidate documents. The technique of constructing a Bloom filter index introduced it enables fast verification of Bloom filters in the same manner as indexing. According to our experiment, it also achieves a lower storage cost than all existing solutions except, where a higher computational cost was exchanged in favor of lower storage. While exhibiting similar communication cost to leading existing solutions, the proposed solution can also be adjusted to achieve maximum speed or high speed with a reasonable storage cost depending on the application. An approach is also described to adapt the scheme to defend against inclusion-relation attacks. Various issues on security and efficiency, such as the effect of long phrases and precision rate, were also discussed to support our design choices.

VII.REFERENCE

[1]. D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in In proceedings of Eurocrypt, 2004, pp. 506–522.

[2]. B. Waters, D. Balfanz, G. Durfee, and D. K. Smetters, "Building an encrypted and searchable audit log," in Network and Distributed System Security Symposium, 2004.

[3]. M. Ding, F. Gao, Z. Jin, and H. Zhang, "An efficient public key encryption with conjunctive keyword search scheme based on pairings," in IEEE International Conference onNetwork Infrastructure and Digital Content, 2012, pp. 526–530.

[4]. F. Kerschbaum, "Secure conjunctive keyword searches for unstructured text," in International Conference on Network and System Security, 2011, pp. 285–289.

[5]. C. Hu and P. Liu, "Public key encryption with ranked multikeyword search," in International Conference on Intelligent Networking and Collaborative Systems, 2013, pp. 109–113.

[6]. Z. Fu, X. Sun, N. Linge, and L. Zhou, "Achieving effective cloud search services: multi-keyword ranked search over encrypted cloud data supporting synonym query," IEEE Transactions on Consumer Electronics, vol. 60, pp. 164–172, 2014.

[7]. C. L. A. Clarke, G. V. Cormack, and E. A. Tudhope, "Relevance ranking for one to three term queries," Information Processing and Management: an International Journal, vol. 36, no. 2, pp. 291–311, Jan. 2000.

[8]. H. Tuo and M. Wenping, "An effective fuzzy keyword search scheme in cloud computing," in International Conference on Intelligent Networking and Collaborative Systems, 2013, pp. 786–789.

[9]. M. Zheng and H. Zhou, "An efficient attack on a fuzzy keyword search scheme over encrypted data," in International Conference on High Performance Computing and Communications and Embedded and Ubiquitous Computing, 2013, pp. 1647–1651.

[10]. S. Zittrower and C. C. Zou, "Encrypted phrase searching in the cloud," in IEEE Global Communications Conference, 2012, pp. 764– 770.

[11]. Y. Tang, D. Gu, N. Ding, and H. Lu, "Phrase search over encrypted data with symmetric encryption scheme,"in International Conference on Distributed Computing SystemsWorkshops, 2012, pp. 471–480.

