A Novel Implementation of Fuzzy Keyword Search over Encrypted Data in Cloud Computing

RANJEETH KUMAR.M
M.Tech Student JNTU Hyderabad A.P. India
D.VASUMATHI
Associate professor CSE JNTU CEH Hyderabad A.P. India

Abstract
Cloud computing is an internet computing in which data is stored and accessed via a remote third party server called cloud, rather than being stored locally on our machine and the resources, software's, and information are provided to users on demand. In cloud computing environment data security is an on-going challenging task, hence the sensitive data has to be encrypted before outsourcing. In existing technique we retrieve the files from the cloud, by searching the keywords on the encrypted data. They are many searching technique which were implemented in the cloud but the disadvantages with these technique supports only exact keyword search. Typical users searching behaviors are happen very frequently these are the drawbacks with the existing system which are not suitable for cloud computing environment and which effects system usability. Our proposed work in this paper concentrates on solving the problems of the user who search the data with the help of fuzzy keyword on cloud. We formalize and solve the problems of effective fuzzy keyword search over encrypted cloud data while maintaining keyword privacy. Using fuzzy search the exact keywords are displayed along with similarity keywords, which solve the problems faced by the cloud users. We show that our proposed solution is secure and privacy preserving, while correctly realizing the goal of fuzzy keyword search.

1. INTRODUCTION

As Cloud Computing becomes prevalent, more and more sensitive information are being centralized into the cloud, such as emails, personal health records, government documents, etc. By storing their data into the cloud, the data owners can be relieved from the burden of data storage and maintenance so as to enjoy the on-demand high quality data storage service. However, the fact that data owners and cloud server are not in the same trusted domain may put our sourced data at risk, as the cloud server may no longer be fully trusted. It follows that sensitive data usually should be encrypted prior to outsourcing for data privacy and combating unsolicited accesses. However, data encryption makes effective data utilization a very challenging task given that there could be a large amount of outsourced data files. Moreover, in Cloud Computing, data owners may share their outsourced data with a large number of users. The individual users might want to only retrieve certain specific data files they are interested in during a given session. One of the most popular ways is to selectively retrieve files through keyword-based search instead of retrieving all the encrypted files back which is completely impractical in cloud computing scenarios. Such keyword-based search technique allows users to selectively retrieve files of interest and has been widely applied in plaintext search scenarios, such as Google search. Unfortunately, data encryption restricts user's ability to perform keyword search and thus makes the traditional plain text search methods unsuitable for Cloud Computing. Besides this, data encryption also demands the protection of keyword privacy since keywords usually contain important information related to the data files. Although encryption of keywords can protect keyword privacy, it further renders the traditional plain text search techniques useless in this scenario.

To securely search over encrypted data, searchable encryption techniques have been developed in recent years. Searchable encryption schemes usually build up an index for each keyword of interest and associate the index with the files that contain the keyword. By integrating the trap doors of keywords within the index information, effective keyword search can be realized while both file content and keyword privacy are well-preserved. Although allowing for performing searches securely and effectively, the existing searchable encryption techniques do not suit for cloud computing scenario since they support only *exact* keyword search. That is, there is no tolerance of minor typos and format inconsistencies. It is quite common that users' searching input might not exactly match those pre-set keywords due to the possible typos, such as Illinois and Ilinois, representation inconsistencies, such as PO BOX and P.O. Box, and/or her lack of exact knowledge about the data. The naive way to support fuzzy keyword search is through simple spell check mechanisms. However, this approach does not completely solve the problem and sometimes can be ineffective due to the following reasons: on the one hand, it requires additional interaction of user to determine the correct word from the candidates generated by the spell check algorithm, which unnecessarily costs user's extra computation effort;

on the other hand, in case that user accidentally types some other valid keywords by mistake (for example, search for "hat" by carelessly typing "cat"), the spell check algorithm would not even work at all, as it can never differentiate between two actual valid words. Thus the drawbacks of existing schemes signifies the important need for new techniques that support searching flexibility, tolerating both minor typos and format inconsistencies.

We focus on enabling effective yet privacy preserving fuzzy keyword search in Cloud Computing. To the best of our knowledge, we formalize for the first time the problem of effective fuzzy keyword search over encrypted cloud data while maintaining keyword privacy. Fuzzy keyword search greatly enhances system usability by returning the matching files when users' searching inputs exactly match the predefined keywords or the closest possible matching files based on keyword similarity semantics, when *exact* match fails. More specifically, we use edit distance to quantify keywords similarity and develop a novel technique, i.e., a wild card-based technique, for the construction of fuzzy keyword sets. This technique eliminates the need for enumerating all the fuzzy keywords and the resulted size of the fuzzy keyword sets is significantly reduced. Based on the constructed fuzzy keyword sets, we propose an efficient fuzzy keyword search scheme. Through rigorous security analysis, we show that the proposed solution is secure and privacy-preserving, while correctly realizing the goal of fuzzy keyword search.

1.1 Motivation

The main goal of this project is to focus on enabling effective yet privacy-preserving fuzzy keyword search in Cloud Computing. To the best of our knowledge, we formalize for the first time the problem of effective fuzzy keyword search over encrypted cloud data while maintaining keyword privacy.

1.2 Problem Definition

Fuzzy keyword search greatly enhances system usability by returning the matching files when users' searching inputs exactly match the predefined keywords or the closest possible matching files based on keyword similarity semantics, when *exact* match fails. More specifically, we use edit distance to quantify keywords similarity and develop a novel technique, i.e., wildcard-based technique and gram-based technique for the construction of fuzzy keyword sets. This technique eliminates the need for enumerating all the fuzzy keywords and the resulted size of the fuzzy keyword sets is significantly reduced. Based on the constructed fuzzy keyword sets, we propose an efficient fuzzy keyword search scheme. Through rigorous security analysis, we show that the proposed solution is secure and privacy preserving, while correctly realizing the goal of fuzzy keyword search.

2 Literature Survey

2.1 Wildcard – Based Technique

In wildcard-based technique all the variants of the keywords have to be listed even if an operation is performed at the same position. Based on the above observation, we proposed to use a wildcard to denote edit operations at the same position. The wildcard-based fuzzy set edit's distance to solve the problems.

For example, for the keyword CASTLE with the pre-set edit distance 1, its wildcard based fuzzy keyword set can be constructed as

SCASTLE, 1 = {CASTLE, *CASTLE,*ASTLE, C*ASTLE...CASTLE*}.

2.2 Gram – Based Technique

Another efficient technique for constructing fuzzy set is based on grams. The gram of a string is a **substring** that can be used as a signature for efficient approximate search. While gram has been widely used for constructing inverted list for approximate string search, we use gram for the matching purpose. We propose to utilize the fact that any primitive edit operation will affect at most one specific character of the keyword, leaving all the remaining characters untouched. In other words, the relative order of the remaining characters after the primitive operations is always kept the same as it is before the operations.

For example, the gram-based fuzzy set SCASTLE, 1 for keyword CASTLE can be constructed as

{CASTLE, CSTLE, CATLE, CASLE, CASTE, CASTL, ASTLE}.

2.3 Searchable Encryption

Traditional searchable encryption has been widely studied in the context of cryptography. Among those works, most are focused on efficiency improvements and security definition formalizations. The first construction of searchable encryption was proposed by Song et al. in which each word in the document is encrypted independently under a special two-layered encryption construction. Goh proposed to use Bloom filters to construct the indexes for the data files. To achieve more efficient search, Chang et al and Curtmola et al. both proposed similar "index" approaches, where a single encrypted hash table

index is built for the entire file collection. In the index table, each entry consists of the trapdoor of a keyword and an encrypted set of file identifiers whose corresponding data files contain the keyword. As a complementary approach, Boneh et al. presented a public-key based searchable encryption scheme, with an analogous scenario to that of. Note that all these existing schemes support only exact keyword search, and thus are not suitable for Cloud Computing.

3 EXISTING SYSTEM AND PROPOSED SYSTEM

3.1 Existing System

This straightforward approach apparently provides fuzzy keyword search over the encrypted files while achieving search privacy using the technique of secure trapdoors. However, this approaches serious efficiency disadvantages. The simple enumeration method in constructing fuzzy key-word sets would introduce large storage complexities, which greatly affect the usability.
For example, the following is the listing variants after a substitution operation on the first character of keyword

CASTLE: {AASTLE, BASTLE, DASTLE, YASTLE, ZASTLE}.

3.2 Proposed System

Main Modules:

Wildcard – Based Technique
Gram - Based Technique
Fuzzy Keyword Set
Searchable Encryption
Construction of Effective Fuzzy Keyword Search in Cloud

3.2.1 Wildcard – Based Technique

In the above straightforward approach, all the variants of the keywords have to be listed even if an operation is performed at the same position. Based on the above observation, we proposed to use an wildcard to denote edit operations at the same position. The wildcard-based fuzzy set edits distance to solve the problems.

For example, for the keyword CASTLE with the pre-set edit distance 1, its wildcard based fuzzy keyword set can be constructed as

SCASTLE, 1 = {CASTLE, *CASTLE,*ASTLE, C*ASTLE, C*STLE, CASTL*E, CASTL*, CASTLE*}.

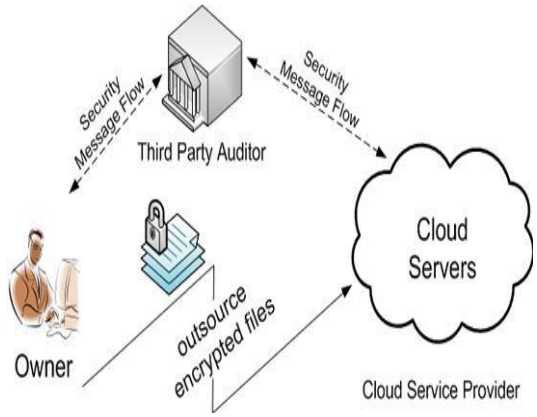
Edit Distance:

a. Substitution
b. Deletion
c. Insertion

a) **Substitution** : changing one character to another in a word;

b) Deletion : deleting one character from a word;

c) **Insertion**: inserting a single character into a word.

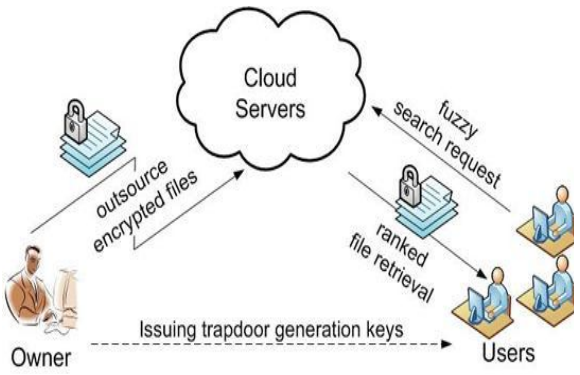


3.2.2 Gram – Based Technique

Another efficient technique for constructing fuzzy set is based on grams. The gram of a string is a **substring** that can be used as a signature for efficient approximate search. While gram has been widely used for constructing inverted list for approximate string search, we use gram for the matching purpose. We propose to utilize the fact that any primitive edit operation will affect at most one specific character of the keyword, leaving all the remaining characters untouched. In other words, the relative order of the remaining characters after the primitive operations is always kept the same as it is before the operations.

For example, the gram-based fuzzy set SCASTLE, 1 for keyword CASTLE can be constructed as

{CASTLE, CSTLE, CATLE, CASLE, CASTE, CASTL, ASTLE}.



3.2.3 Searchable Encryption

Traditional searchable encryption [2]–[8], [10] has been widely studied in the context of cryptography. Among those works, most are focused on efficiency improvements and security definition formalizations. The first construction of searchable encryption was proposed by Song et al. [3], in which each word in the document is encrypted independently under a special two-layered encryption construction. Goh [4] proposed to use Bloom filters to construct the indexes for the data files. To achieve more efficient search, Chang et al. [7] and Curtmola et al. [8] both proposed similar "index" approaches, where a single encrypted hash table index is built for the entire file collection. In the index table, each entry consists of the trapdoor of a keyword and an encrypted set of file identifiers
Whose corresponding data files contain the keyword. As a complementary approach, Boneh et al. [5] presented a public-key based searchable encryption scheme, with an analogous scenario to that of [3]. Note that all these existing schemes support only exact keyword search, and thus are not suitable for Cloud Computing.

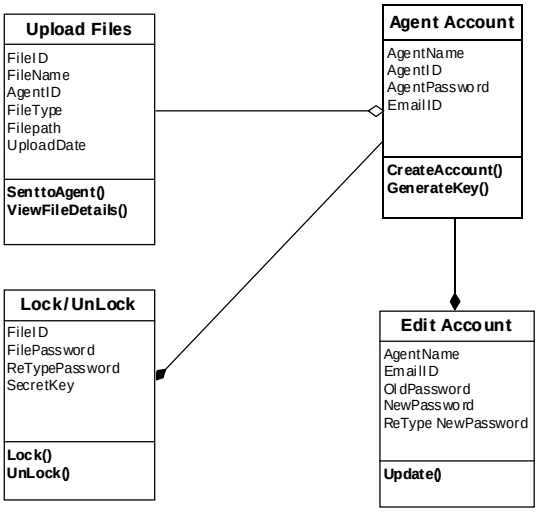
3.2.4 Construction of Effective Fuzzy Keyword Search in Cloud

The key idea behind our secure fuzzy keyword search is two-fold:

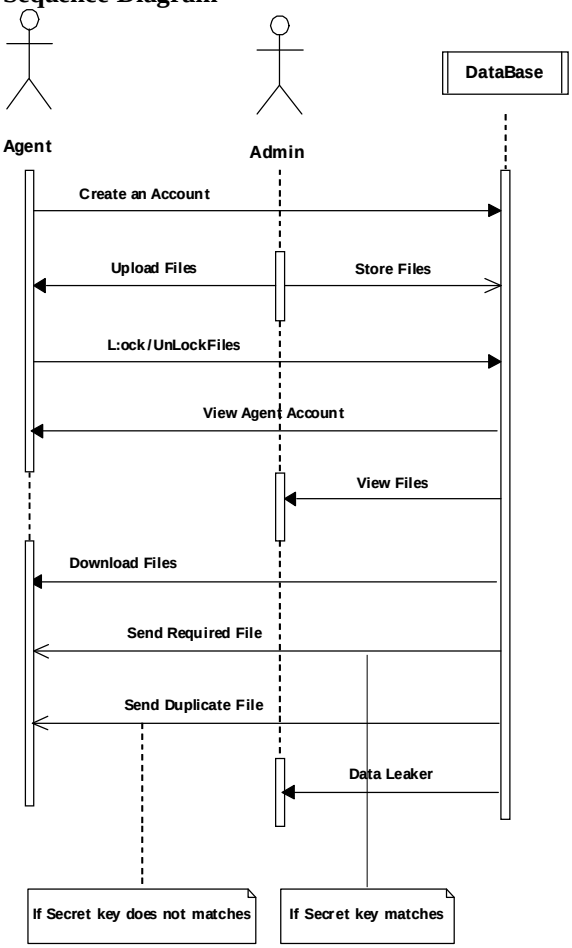
1) Building up fuzzy keyword sets that incorporate not only the exact keywords but also the ones differing slightly due to minor typos, format inconsistencies, etc.
2) Designing an efficient and secure searching approach for file retrieval based on the resulted fuzzy keyword sets.

4 SYSTEM DESIGN:

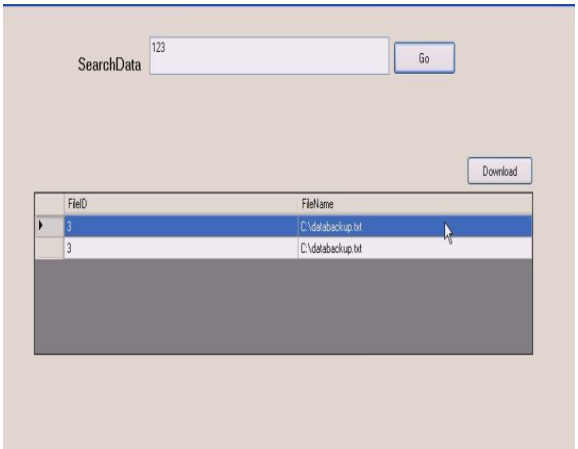
Class Diagram



Sequence Diagram



IMPLEMENTATION SCREEN SHOT



5 CONCLUSION & FUTURE WORK

In this project, for the first time we formalize and solve the problem of supporting efficient yet privacy-preserving fuzzy search for achieving effective utilization of remotely stored encrypted data in Cloud Computing. We design an advanced technique (i.e., wild card-based technique) to construct the storage-efficient fuzzy keyword sets by exploiting a significant observation on the similarity metric of edit distance. Based on the constructed fuzzy keyword sets, we further propose an efficient fuzzy keyword search scheme. Through rigorous security analysis, we show that our proposed solution is secure and privacy-preserving, while correctly realizing the goal of fuzzy keyword search.

Future work is on security mechanisms that support search semantics that takes into consideration conjunction of keywords, sequence of keywords, and even the complex natural language semantics to produce highly relevant search results and search ranking that sorts the searching results according to the relevance criteria.

6 REFERENCES

1) Google, "Britney spears spelling correction," Referenced online at http://www.google.com/jobs/britney.html,June 2009.

2) M. Bellare, A. Boldyreva, and A. O'Neill, "Deterministic and efficiently searchable encryption," in Proceedings of Crypto 2007, volume 4622 of LNCS. Springer-Verlag, 2007.

3) D. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in Proc. of IEEE Symposium on Security and Privacy'00, 2000.

4) E.-J. Goh, "Secure indexes," Cryptology ePrint Archive, Report 003/216,2003, http://eprint.iacr.org/.

5) D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in Proc. of EUROCRYP'04, 2004.

6) B. Waters, D. Balfanz, G. Durfee, and D. Smetters, "Building an encrypted and searchable audit log," in Proc. of 11th Annual Network and Distributed System, 2004.

7) Y.-C. Chang and M. Mitzenmacher, "Privacy preserving keyword searches on remote encrypted data," in Proc. of ACNS'05, 2005.

8) R. Curtmola, J. A. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: improved definitions and efficient constructions," in Proc. of ACM CCS'06, 2006.

9) D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data," in Proc. of TCC'07, 2007, pp. 535–554.

10) F. Bao, R. Deng, X. Ding, and Y. Yang, "Private query on encrypted data in multi-user settings," in Proc. of ISPEC'08, 2008.

11) C. Li, J. Lu, and Y. Lu, "Efficient merging and filtering algorithms for approximate string searches," in Proc. of ICDE'08, 2008.

14) J. Feigenbaum, Y. Ishai, T. Malkin, K. Nissim, M. Strauss, and R. N.Wright "Secure multiparty computation of approximations," in Proc. of ICALP'01.

15) R. Ostrovsky, "Software protection and simulations on oblivious rams," Ph.D dissertation, Massachusetts Institute of Technology, 1992.

16) V. Levenshtein, "Binary codes capable of correcting spurious insertions and deletions of ones," Problems of Information Transmission, vol. 1,no.1, pp. 8–17, 1965.