International Journal of Computer Applications (0975 – 8887)
Volume 106 – No.7, November 2014

Implementation of Keyword Search Considering User's Preference in Encrypted Data in Cloud Computing

R Kalpana
K J Somaiya, COE
Vidyavihar, Mumbai

Manish M Potey
K J Somaiya, COE
Vidyavihar, Mumbai

ABSTRACT

Cloud Computing is very popular and used in the every field let it be for usage of software or as a platform or as an infrastructure. Cloud computing is now an important thing around which the technologies revolve. Cloud computing has made everything available in just a click away format. Heavy data is stored using storage as a service or a huge amount of data resides on cloud for the various transactions to be done. When such a huge dependence is done on cloud computing it's obvious for the data to be stored in encrypted format. But when the data is stored in encrypted format, then the option of doing search on plaintext data is not possible. Therefore search on encrypted data is must. On addition, if the user is able to give his own preference then the search will be more efficient. Thus in this paper we will tell how the keyword search considering user's preference is implemented. Whenever an owner stores a file in cloud an index for that is automatically generated and the file is encrypted and stored in the cloud. When a user wants to search from the data stored in the cloud in encrypted format, he can give the list of keywords along with his preferences, i.e. which keyword has higher or lower prefence.Corresponding to that preference the application retrieves the file from cloud and in the ranking order as of the preference.in our implementation all the files are stored in owncloud[13]

General Terms

Information Retrieval, Cloud computing.

Keywords

Searchable Encryption, User Preference, Ranking, owncloud

1. INTRODUCTION

The Cloud computing is a major technology used now days. A huge amount of data is stored as well as used for various transaction purposes. So to protect data privacy, by default the data is stored in encrypted format in cloud, which does not allow searching on the plain text. So the search on encrypted data is must. There are various searchable encryption techniques which allow searching on encrypted text. Our paper explains how to do searching on the encrypted text and allowing user to give his preference while searching to generate more accurate results according to preference for the specific user. The need for the Keyword search considering users preference (KSCUP) is required because the naive search output without preference consideration will transmit all the matching files and will cause network congestion. The KSCUP allows multikeyword search and allows the user to give their preferences for the keyword to retrieve selected files. In our implementation all the files are stored in owncloud, which is file sync and share solution that is as easy to use as for the consumer-grade products, but it is hosted in owner data center, on owner servers, using owner storage. ownCloud integrates into the IT infrastructure, and it delivers file sharing services that in compliance to the data security and policies. ownCloud is the preferred file sharing solution for enterprises everywhere.

2. RELATED WORK

There has been a lot of work been done in searchable encryption. During last several years, searchable encryption (SE) [2]-[11] has been evolved in pursuit of search over encrypted data under different applications. Song et al. [3] proposed the first working scheme of searchable encryption. Goh et al. [4] presented a scheme that supports secure indexing over encrypted data. Boneh et al. [5] proposed the first searchable encryption scheme based on public key. Water et al. [7] proposed a scheme to fulfil searchable audit log. Golle et al. [8] developed two searchable encryption schemes to realize conjunctive keyword search. Wang et al. [9] and Cao et al. [11] investigated secure ranked keyword search on single keyword and multiple keywords over encrypted data respectively. Shiet al. [10] presented their method to realize multi-dimensional range query over encrypted data. However, most of the existing SE works missed users' preferences when performing search. For schemes [2], [8], [10] they implement the flexible search, they support only "Boolean keyword search" and give limited attention to the relevance between files and a query. For schemes that implement ranked keyword search, they either just support single keyword search [9] or may return inaccurate results [11]. Most of the existing works ignore the user's preference, easily leading to the following drawbacks.

- A user who does not have any pre-understanding about the encrypted data has to endure the labor- intensive task of manually picking out their interested files.
- The naive search output without preference consideration will easily cause network congestion because of the transmission for all the matching files.

In short, the absence of preferred search with privacy preserved and flexible search query support is still a typical shortage in existing SE schemes. Thus our paper fulfils the shortage by implementing the same.

Leubner et al. [13] proposed prioritization can be explained as the subspace preferences in vector space model. Koutrika et al. [1] showed preference model. Chomicki et al. [14] explained the framework for calculating preferences. Kiessling et al. [15] showed partial order meaning for preferences. But mostly existing search with preference is not available in ciphertext.

3. SYSTEM MODEL

Our system allows the admin to have a control on the files stored in the owncloud.Data owner can upload the files in the cloud i.e. C= $\{F_1, F_2 \ldots F_n\}$ and during which an index file *I* is created for each owner and the file is encrypted and stored over the cloud and it allows authorized data users to do search on them. Data users can give a multikeyword query Q along with the preferences P .When receiving the query and preference, index file is scanned and matching files are retrieved from the cloud according to the preferences. Thus this saves the time of the user from selecting the required file from a huge massive storage.

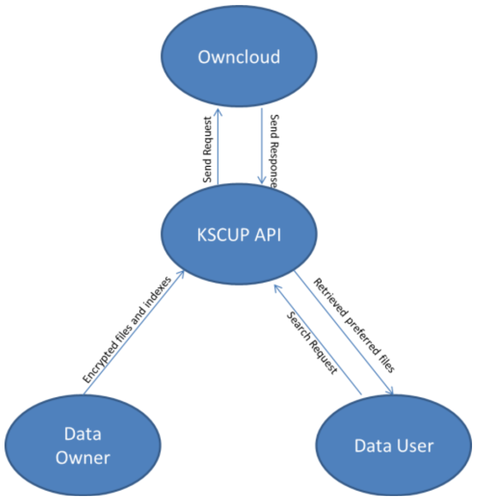


Figure 1: System Model

4. EXPERIMENTAL RESULTS

All functions are performed on Intel i5-480M processor using MYSQL as a backend for database and JSP and Java for front end programming. Three tables are used First, to store the details of data owner and data user. Second to store keys to generate index and retrieve files.Third,to keep a track of the files downloaded.WAMP server is used for owncloud.

4.1 The Algorithm for Storing the Files in the Cloud

Step 1: Data owner (DO) will write a file(supported formats are doc,docx,pdf and text files)

Step 2: DO will login into the KSCUP application

Step 3: After login, he will select the file to be uploaded. If the file is other than supported format an error will generate.

Step 4: After selecting the upload, the function will check

whether it is doc, pdf or text files.

Step 5: Once it identifies the extension of the file, the appropriate function is used to extract the words from the file.

Step 6: Once the words are extracted, these are sent to create an index for each word in the file.

Step 7: To create an index a hash map<string, integer> is used which is encrypted and stored in an index file and a random key is generated for the hash map which is stored in database.

Step 8: The actual file is encrypted and stored in owncloud along with the index file.

4.2 The Algorithm to Search by Giving User Preference

Step 1: To search a keyword. The Data User (DU) has to login into application.

Step 2: After authorized login, DU can give the keywords which he wants to search. The keywords are followed by the preference.

Step 3: The preference is processed and the data i.e. the keyword and preference is sent to the cloud.

Step 4: Then as per the preference the higher preferred keyword is searched first.

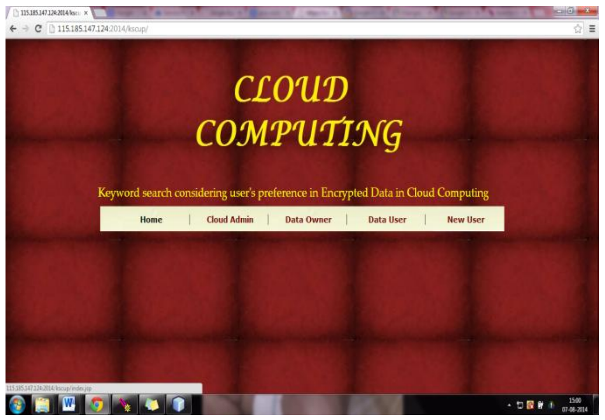
Step 5: Then for each file in the cloud, firstly the random key which is stored in database is retrieved and the hash map value is retrieved and checked with the keyword entered by the user.

Step 6: If the hash map value matches with the keyword then the term frequency and inverse document frequency is updated.

Step 7: Once all the files are scanned then the same process is repeated with the remaining keyword.

Step 8: Once all the keywords are searched then matching preferred files are displayed to the user with the option of downloading the same

Now the server is started and home screen is displayed first

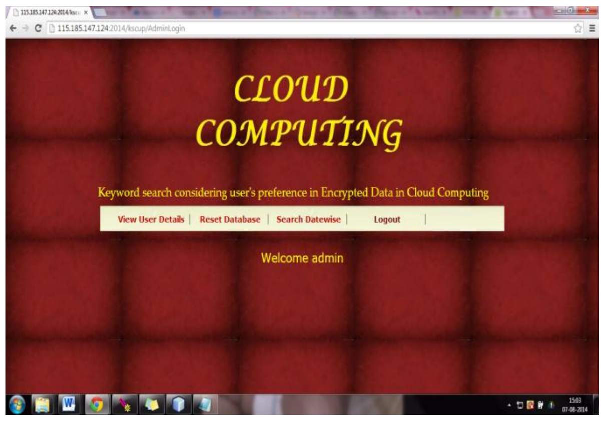

115.185.147.124:2014/kscup/

CLOUD COMPUTING

Keyword search considering user's preference in Encrypted Data in Cloud Computing

Figure 2 : Home Screen

It is the application home screen where admin,data owner and data user can login as well as new user can be registered.


115.185.147.124:2014/kscup/AdminLogin

CLOUD COMPUTING

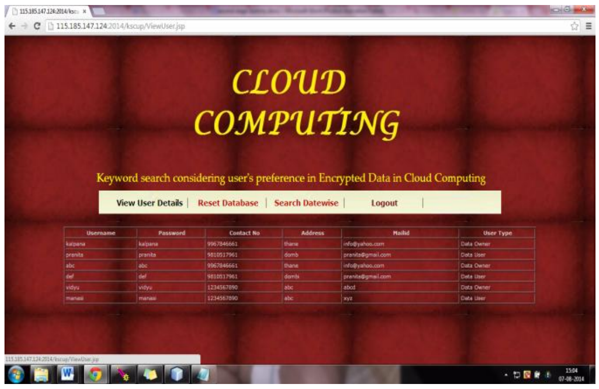
Keyword search considering user's preference in Encrypted Data in Cloud Computing

View User Details | Reset Database | Search Datewise | Logout |

Welcome admin

Figure 3: Post Admin Login

Once admin does the login then he can view the entire DO and DU details, can reset the database.


CLOUD COMPUTING

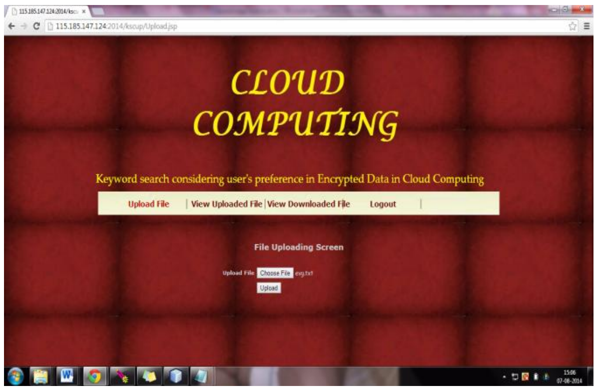
Keyword search considering user's preference in Encrypted Data in Cloud Computing

View User Details | Reset Database | Search Datewise | Logout

Username	Password	Contact No	Address	Mailid	User Type
kalpana	kalpana	9967846661	thane	info@yahoo.com	Data Owner
pranita	pranita	9810517961	domb	pranita@gmail.com	Data User
abc	abc	9967846661	thane	info@yahoo.com	Data Owner
def	def	9810517961	dombi	pranita@gmail.com	Data User
vidyu	vidyu	1234567890	abc	abcd	Data Owner
manasi	manasi	1234567890	abc	xyz	Data User

Figure 4: View user details

All DO and DU details are shown


CLOUD COMPUTING

Keyword search considering user's preference in Encrypted Data in Cloud Computing

Upload File | View Uploaded File | View Downloaded File | Logout

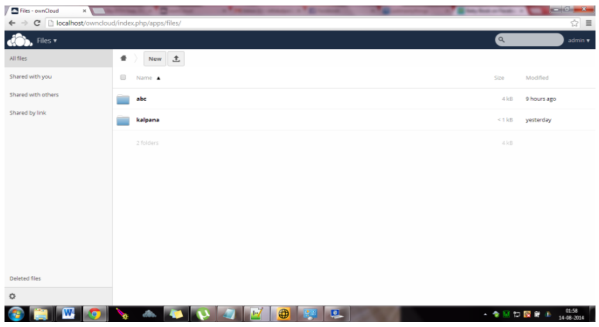
File Uploading Screen

Upload File Choose File

Upload

Figure 5: Post Data Owner Login and Uploading a file

Once data owner logins into system then he can upload a file into cloud and can check which of his files are downloaded and by which data users and can keep a check of the files uploaded by himself.


Files - ownCloud

localhost/owncloud/index.php/apps/files/

Name	Size	Modified
abc	4 kB	9 hours ago
kalpana	< 1 kB	yesterday
2 folders	4 kB	

Figure 6: Files stored in owncloud for each Data owner

All the files uploaded by the data owner are stored in the cloud under the data owner name.

5. COMPARISION WITH OTHER SEARCH METHODS

When KSCUP is compared with other search techniques in cloud, KSCUP gives an advantage of having multikeyword, search along with user's preference which other technique does not provide. But the size of index file is very huge if the number of files is increased, which turns out to be a disadvantage. Although the search efficiency is increased but the processing of files and creating indexes is a bottleneck for efficiency. Thus if a better way of storing indexes is implemented then the overall efficiency would be very much higher. Thus in comparison to other works, the index building and time of search index remains the same but other works does not provide flexible query, preferred search and accurate search result.

6. CONCLUSION

In this paper, as an initial attempt, we propose and solve the problem of doing search over encrypted data and giving importance to user's preference and giving a ranked order of search. It supports multikeyword and displays the ranked order according to term weighting concept. We show that existing framework does only searching and generates random results which make a hectic task for user to manually pick up the files. The query privacy is maintained.

7. FUTURE WORK

This implementation efficiency can be further enhanced by developing a better way of creating indexes. Although creating index is one time cost but when the size of the files and number of files increases the overall efficiency is reduced.

8. REFERENCES

[1] G. Koutrika and Y. Ioannidis. Personalized queries under a generalized preference model. In Proc. of ICDE, 2005

[2] D. Boneh and B. Waters.: Conjunctive, subset, and range queries on encrypted data. in Proc. of TCC, 2007

[3] D.Song, D.Wagner, and A.Perrig. Practical techniques for searches on encrypted data. In Proc. of IEEE S&P, 2000

[4] E.Goh. Secure indexes. Cryptology ePrint Archive, 2003, http://eprint.iacr.org/2003/216

[5] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano. Public key encryption with keyword search. In Proc. of EUROCRYPT, 2004

[6] Y.-C. Chang and M. Mitzenmacher. Privacy preserving keyword searches on remote encrypted data. In Proc. of ACNS, 2005

[7] B. Waters, D. Balfanz, G. Durfee, D.K. Smetters. Building an encrypted and searchable audit log. In Proc. of NDSS, 2004

[8] P. Golle, J. Staddon, and B. Waters. Secure conjunctive keyword search over encrypted data. In Proc. of ACNS, 2004

[9] C. Wang, N. Cao, J. Li, K. Ren, and W. Lou. Secure ranked keyword search over encrypted cloud data. in Proc. of ICDCS, 2010

[10] E. Shi, J. Bethencourt, T. Chan, D. Song, and A. Perrig. Multidimensional range query over encrypted data. In Proc. of IEEE S&P, 2007

[11] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou. Privacy-preserving multikeyword ranked search over encrypted cloud data. In Proc. of INFOCOM, 2011

[12] A. Leubner and W. Kiessling. Personalized keyword search with partial order preferences. In SBBD, 2002

[13] doc.owncloud.org/server/6.0/user_manual

[14] J. Chomicki. Preference formulas in relational queries. ACM Transaction Database Systems, 28(4), 2003

[15] W. Kiessling. Foundations of preferences in database systems. In Proc.of VLDB, 2002