Performance Evaluation of multi keyword ranked search schema called BDMRS-CM & EDMRS-BM in Cloud computing

[1]Pawan Kumar Tanwar, [2]Ajay Khunteta, [3]Vishal Goar
[1]Research Scholar, [2]Professor, [3]Assistant Professor
[1]Department of Computer Science, Poornima University, Jaipur,
[2]Department Computer Science, Poornima University, Jaipur
[3]Department of Computer Application, Govt. Engineering College, Bikaner
[1]pktbkn@gmail.com, [2]khutetaajay@poornima.org, [3]dr.vishalgoar@gmail.com

ABSTRACT

Nowadays, Everything can be searched through the cloud space. Therefore, this article is also dealing with the cloud computing, where searching of information and preservation of privacy is key area of concern. Henceforth taking multi-keyword ranked search with dynamic updation as a dimension of information searching, have been selected it as a research area. Moreover the searching is restricted to single keyword only. Therefore, we have taken the concept of multi keyword ranked searching. One more thing is that efforts have not been made regarding dynamic updation (insertion and deletion etc. of documents) previously. To cover up the dynamic updation part the schemes BDMRS-CM (Basic Dynamic Multi-Keyword Ranked Search scheme in the Known Ciphertext Model) by using the secure kNN algorithm and the EDMRS-BM (Enhanced Dynamic Multi-Keyword Ranked Search scheme in the Known Background Model) were designed and their performance have been evaluated and analyzed.
Keywords: cloud computing, dynamic updation, multi-keyword

INTRODUCTION

Searching of information or keyword in the cloud space is not so easy. Further types of searching are single keyword and multi keyword searching. A lot of works have been done on single keyword searching but more work has to be done on multi keyword searching.

We have proposed two new schemes for the fulfillment of dynamic updation part of the proposed research work. The schemes are BDMRS-CM (Basic Dynamic Multi-Keyword Ranked Search scheme in the Known Ciphertext Model) by using the secure kNN algorithm and the EDMRS-BM (Enhanced Dynamic Multi-Keyword Ranked Search scheme in the Known Background Model).

For performance analysis we have performed the simulation for different parameters. We have checked the performance by using various parameters like time, cost and number of files for index, trapdoor and query [1][2][3].

METHOD OF EXPERIMENT

Implementation of the scheme has been done through the C++ language and the OS is windows 7. Finally we have checked the efficiency on number of plain text files. The checking includes (a) The precision of searching on distinguished level of privacy and (b) The efficiency of the construction of index, generation of trapdoor, update and search. Almost all the results of experiment are got through a processor of Intel Core to Duo 2.96 GHz. The searching efficiency is checked with a server where two processors having twelve cores of processor with supporting of twenty four threads parallel.[4][5]

PERFORMANCE EVALUATION

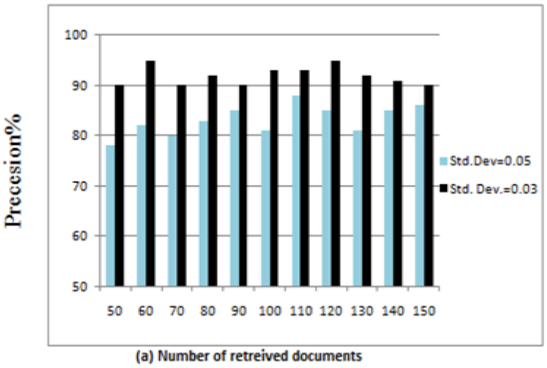
A. Privacy and Precision

The searching precision of the schema is influenced by blank keywords quantity in EDMRS-BD scheme. The definition of precision is Pk = k'/k where k' is the quantity of actual top-k files in the retrieved k files. The EDMRS-BM scheme is supposed to get greater precision, if a lower std. dev. (σ) is applied for dynamic variable ∑εv, and in reverse also. The outputs are demonstrated in figure 1(a).

Talking about EDMRS-BD schema, few expressions (phantom) are joined with vectors of index for obscuring the calculation of relevancy result; therefore the machine (server) of cloud could not do the identification of keywords by the analysis of the TF distributions of particular keywords. The uncertainty for relevancy result is done from the quantification of "rank privacy", described as:

$$Pk' = \sum |ri - ri'| / k2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

Where ri denotes ranking position for files in the extracted uppermost (k) files as well as ri' shows original ranking position for aggregated results of ranking. Greater the rank privations shows greater safeguard for schema, that is demonstrated in figure 1(b).



(a) Number of retreived documents

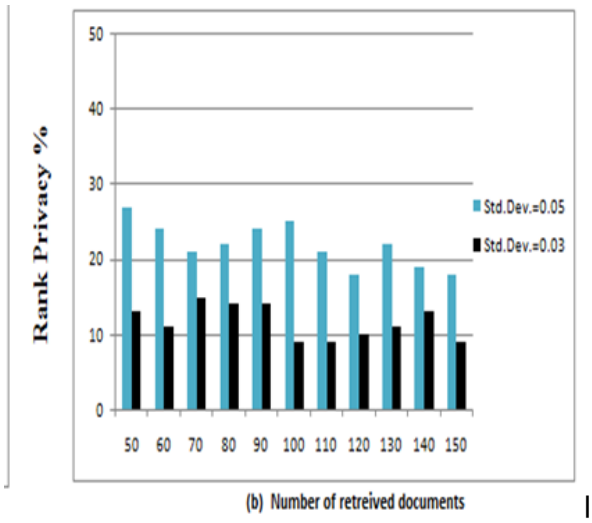


Figure - 1. (a) precision and (b) ranking privacy of searching with different std. deviation

In the proposed schema the users of data can complete distinguished requirements on searching of privations as well as preciseness from adjustment of the std. dev.(σ), that could be used for parameter of balancing.

We have compared this work from earlier proposed schema that gains high searching productivity. The BDMRS-CM schema gains searching result by definite computation of query as well as file vector. Therefore uppermost (k) searching preciseness for BDRMS-CM schema is hundred percent though for a similarly basis multi-keyword rank searching schema, the fundamental schema experiences the shrinkage of precision l as a result of cluster formation for sub-vector meanwhile formation of index. Preciseness testing for the fundamental schema is demonstrated through table (1).For every testing, five key-words have selected for input in a random way and the precision of extracted top hundred results is taken under observation. Testing is reciprocated sixteen number of time hence moderate precision received is ninety one percent. [6] [7] [8]

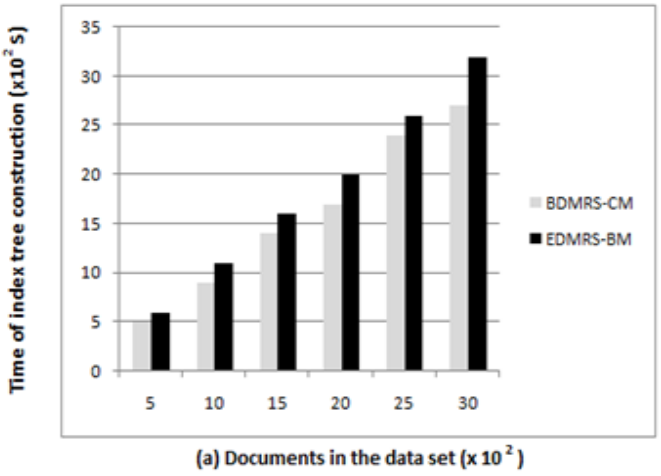
Test No.	Precision	Test No.	Precision
1	88%	9	96%
2	94%	10	86.7%
3	97%	11	87.5%
4	100%	12	100%
5	85%	13	82.3%
6	89%	14	100%
7	89%	15	100%
8	96%	16	71.5%

Table-1 Precision Test of Basic Scheme

B. Efficiency

(1) Construction of Indexed Tree – Method of indexed tree formation for aggregation of files (F) combines 2 chief ways – (a) Construction of a encryption-less (KBB) tree lied on collection of files F and (b) encrypting the index tree through operation of splitting of two multiplications of matrix(n x n). The structure of index is formed through traversal (post-order) of a tree lied upon files bunch F as well as O(m) links are created meanwhile traversing. Every link creation of the indexed vector grasps O(n) timing and product of two matrices (n x n) grasps O(n2) timing. In the totality complexity of time of construction of indexed tree given by O(nm2). Timing overhead of construction for indexed tree is basically depending upon cardinal value for the collection of files F as well as quantity of key-word into the glossary (W). Figure-2 demonstrates timing overhead for construction of indexed tree has more or less continuous along amount of bunch of files as well as proportionate with the quantity of key-words in the glossary. For the reason of expansion of dimension, the indexed tree formation for EDMRS-BM schema is little bit extra time taking in comparison of BDMRS-CM schema. However construction of indexed tree takes comparatively more timing in owner (data) part and notable that it is onetime performance.

Besides that equal paired tree (balance binary tree) having O (n) complexity of space as well as each link storing 2 vector of m dimension and complexity of space on indexed tree has O(nm). As Table-(2) shows at the time files bunch is definite (n is equal to one thousand), space utilization for indexed tree has derived from volume of glossary. [9] [10] [11]



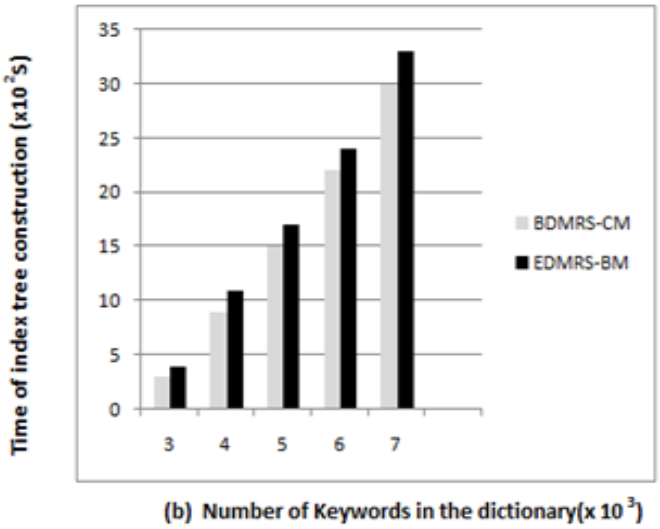


Figure-2- Timing overhead about construction of indexed tree: [a] distinguished volume for file bunch (definite glossary, m is equal to four thousand (b) distinguished volume of glossary when file bunch n is equal to one thousand.

Size of Dictionary	1000	2000	3000	4000	5000
BDMRS-CM(MB)	72	145	221	295	366
EDMRS-BM(MB)	94	167	242	316	389

Table – (2) - Space taken by indexed tree in memory

C. Generation of Trapdoor

The formation of trapdoor overheads a vector breaking procedure as well as product of two matrix (n x n), thus the complexity of time is O (m2), as shown in figure 3(a). A search request generally consists of some key-words. Figure 3(b) demonstrates about quantity of queried key-words having little impact upon cost for trapdoor formation for definite size glossary (dictionary). For the reason of dimensional expansion, timing overhead of EDMRS-BM schema is a bit greater than BDMRS-CM schema.

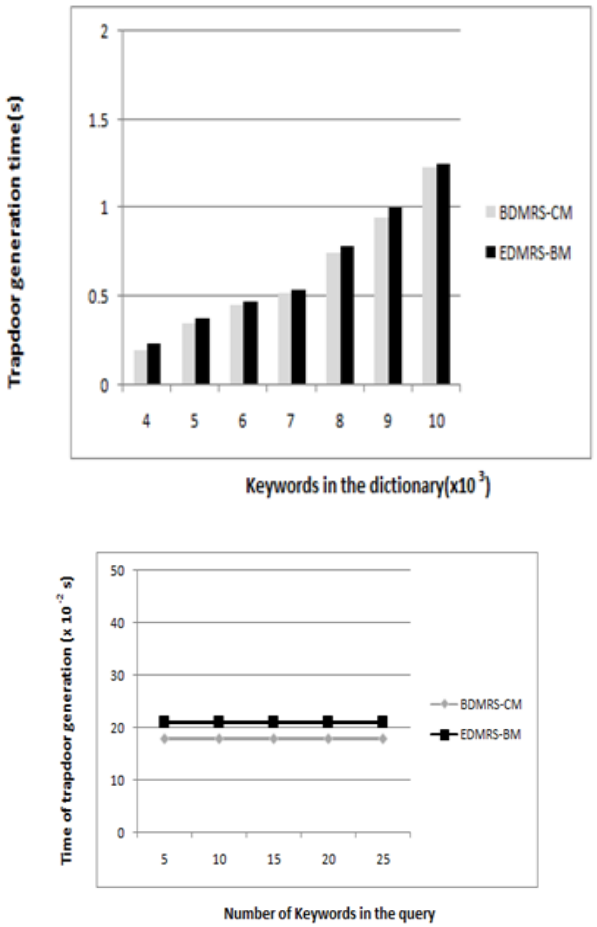


Figure-3 Timing cost on trapdoor creation (a) Distinguished volume (size) of glossary (dictionary) when definite quantity of query key-word (t = ten) (b) Distinguished quantity of query key-word when definite glossary (dictionary) m=four thousand

D. Efficiency of Search

In searching procedure, in case, result of relevance upon link u is greater than lowest result of relevance in output data, the server machine evaluates for child of link; otherwise return back. Therefore, numbers of links are not approached meanwhile actual search. The quantity of last end links those containing single or greater than one key-word in query is denoted by θ. Basically θ is greater than quantity of essential files k, but very few from cardinal number of file bunch n. In equal paired tree (balance binary tree), tallness of index is managed as log(n) and complexity of relevancy result computation defined as O(m).Therefore, the timing complexity for searching denoted as O (θ m log n). It is observed that actual searching timing has lower from θmlogn. As a result of (a) Numerous last end links include the asked key-word has not traversed as per the proposed formula (algorithm) of searching (b) the approaching way for distinguished last end links (leaf node) sharing the bilateral visited paths. Moreover, simultaneous implementation of searching method could enhance the productivity very much.

The efficiency of searching for suggested schema has been tested upon a machine supporting twenty four instance in parallel. Efficiency of searching is checked through one, four, eight and sixteen instances (threads) one by one. Efficiency of searching for suggested schema with other schema has been compared by us. In the implementation of other scheme four thousand keywords were broken into fifty stairs. Therefore, every stair having eighty key-word. As per other schemes greater stair the query key-word lies, greater the efficiency of searching. Ten key-word has been selected from first stair for comparing searching performance.

Figure- 4 demonstrates searching efficacy for suggested schema increased more while increased the quantity of instance (threads) from one to four. Although, while we have maximize the instance (threads) the efficacy of searching has not maximized up to the mark. The suggested algorithm for searching could be applied laterally to enhance efficacy of searching.

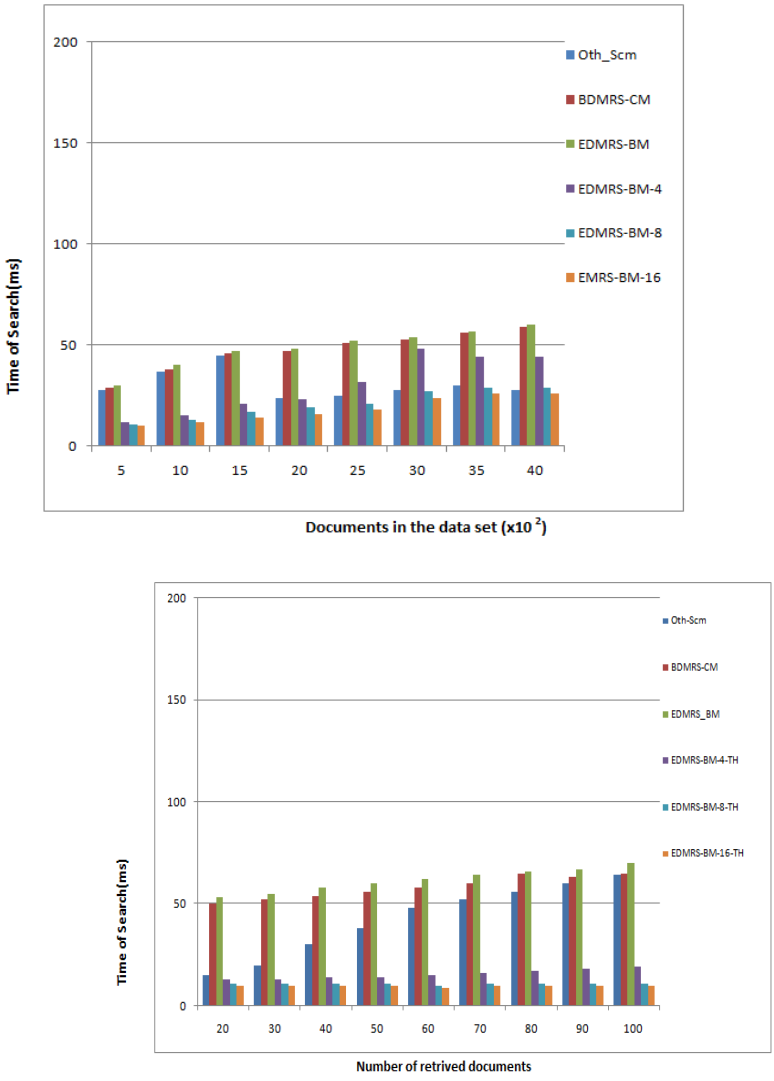


Figure -4-The efficacy of searching by ten key-word of concern as : (a) Distinguished volume of collection of file along like glossary(dictionary), m=four thousand (b) Distinguished quantity of received files along like file bunch and glossary (dictionary), n=one thousand , m=four thousand.

E .Efficiency of updation

For updating a last end link (leaf node), the possessor of data has to modify (update) log n links. As an encipher procedure is involved here for vectors of index on every link, that yields O (m2) timing, timing multiplicity (complexity) for updation procedure is therefore O (m2 log n). Timing overhead of deletion of a file has also been described. Figure 5(a) demonstrates that while the volume of glossary (dictionary) is definite then deletion of file yields near to log time along the volume of the total files in the collection. Figure 5(b) demonstrates the modify (updation) timing is proportionate to volume of the glossary (dictionary) while the bunch of the file is definite. Moreover complexity of space for every link is O(m). So the complexity of space for the transmission container for update the file is O (m log n).

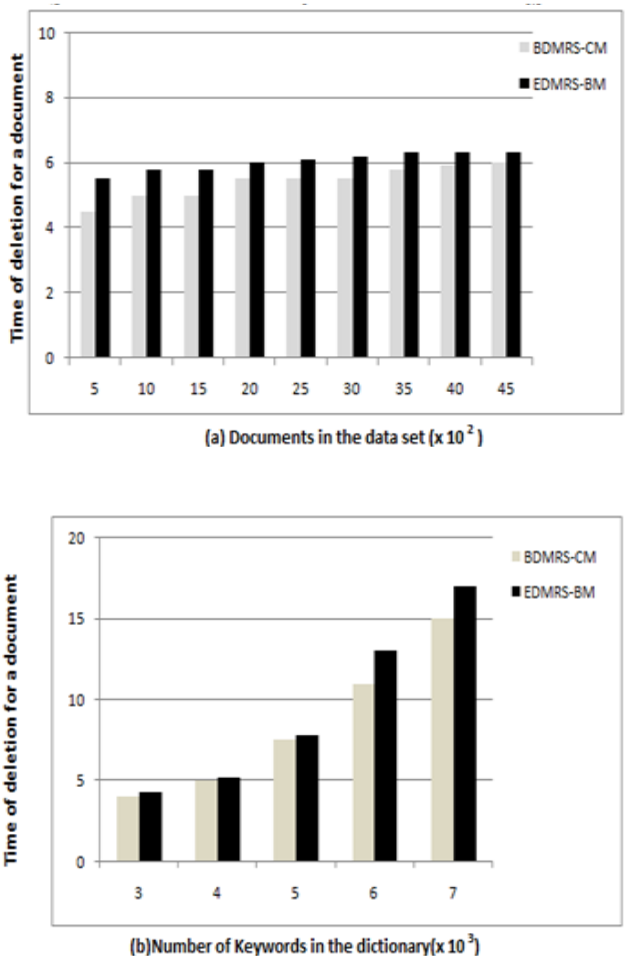


Figure-5 – Timing overhead for deleting a file: (a) distinguished volume (size) of file bunch along the alike glossary (dictionary) size m= four thousand (b) Alike file bunch along distinguished volume (size) of glossary (dictionary) n=one thousand.

CONCLUSION

It is concluded that the multi key-word ranked search scheme upon cloud data (MKRSCD) and MKRSCD-CM (multi keyword ranked search scheme for cloud data in known ciphertext model) both the schema are emphasizing over the searching of keyword and preservation of privacy part very efficiently and effectively but later on two other schema proposed viz BDMRS-CM (Basic Dynamic Multi-Keyword Ranked search scheme in the Known Ciphertext Model) and EDMRS-BM(Enhanced Dynamic Multi-Keyword Ranked search scheme in the Known Background Model) emphasizes not only on multi keyword ranked search and preservation of privacy but also opens the way for dynamic update (insertion and deletion etc.) .

Here an efficient, dynamic and secure search schema is suggested, that supporting accurate multi keyword ranked search and the dynamic insertion and deletion of files. We have constructed a KBB (keyword balance binary) tree as the index and proposed a (greedy depth first search) algorithm to get more efficacy than sequential (linear) searching. Moreover the lateral (parallel) searching could be applied for reducing the timing overhead. The safeguard of the schema is preserved against 2 threat models by applying the secure kNN algorithm. Method of experiment shows the efficiency of the proposed schema [12].

The results of experiments and analysis shows that the designed schemes could enable the multi keyword rank search efficiently and effectively. We have also investigated some more enhancements of rank search methods including the support of additional search semantics like term frequency (TF) x inverse file frequency (IDF) as well as dynamic data operations to fulfill the requirement of dynamic updation feature. To fulfill the complete research work a lot of experiments need to be done for dynamic

REFERENCES

[1] Narendra K. and Narsimhareddy Gkv. , “Dynamic Multi-Keyword Ranking Scheme on Encrypted Cloud Data”, International Journal of Innovative Technology and Research, Volume No.4, Issue No.4, June – July 2016.

[2] Gomathi M. and Seenivasan D., “Dynamic multi-keyword rank scheme using Top key over encrypted cloud data”, International Research Journal of Engineering and Technology (IRJET), Volume: 03 Issue: 04 | April-2016.

[3] Narayankar Ajaykumar, Rathod Gajanan, Londhe Sanket , Wankhade Ashish and Ansari M.A., “Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data”, International Journal of Innovative Research in Computer and Communication Engineering, Vol. 4, Issue 2, February 2016.

[4] Neeshima P.P., Hegde Pavitra Shankar, P. Poojashree and Pallavi G.B, “A multi keyword ranked search technique with provision for dynamic update of encrypted documents in cloud”, International Journal of Computer Engineering and Applications, Volume X, Issue III, March 16.

[5] Karthick K.S. and Deepa P , “A Secure and Dynamic Multi-keyword Ranking Search On Encrypted Cloud Data using GDFS”, International Journal On Advanced Computer Theory And Engineering (IJACTE), Volume -5, Issue -2, 2016

[6] HARIKA HAMPI K. S., LAKSHMI K. and PREM KUMAR S., "A Secure and Dynamic Multi Keyword Ranked Search Scheme Over Encrypted Cloud Data", International Journal of Innovative Technologies, Volume.04, Issue No.08, July-2016, Pages: 1406-1411

[7] Saravanan K.S.and Karthika S., "A Secure and Dynamic Multi-keyword Ranked Search Scheme over Encrypted Cloud Data", International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 2, February 2016

[8] Metkari Siddheshwar S. and Sonkamble S.B., "Multi-keyword Ranked Search Over Encrypted Cloud Data Supporting Synonym Query", International Journal of Science and Research (IJSR), Volume 5 Issue 6, June 2016

[9] Jain Purva and Banubakode Abhijit, "A Review Paper on Multi keyword Ranked Search on Encrypted Cloud Data", IOSR Journal of Computer Engineering (IOSR-JCE), PP 28-32, 2015

[10] Xia Zhihua, Wang Xinhui, Sun Xingming and Wang Qian , "A Secure and Dynamic Multi-keyword Ranked Search Scheme over Encrypted Cloud Data", IEEE Transactions On Parallel And Distributed Systems, Vol. 1, p.p.1-13, 2015

[11] Strizhov Mikhail, "Towards a Practical and Efficient Search over Encrypted Data in the Cloud", IEEE International Conference on Cloud Engineering, Vol. 15, p.p. 496-498, 2015

[12] Chen Chi, Zhu Xiaojie, Shen Peisong, Hu J., Guo S., Tari Z.and Zomaya Albert Y. "An Efficient Privacy Preserving Ranked Keyword Search Method", IEEE Transactions on Parallel and Distributed Systems, Vol. 1, p.p. 1-14,2015