International Journal of Scientific Research and Management (IJSRM)
||Volume||5||Issue||12||Pages||7610-7617||2017||
Website: www.ijsrm.in ISSN (e): 2321-3418
Index Copernicus value (2015): 57.47, (2016):93.67, DOI: 10.18535/ijsrm/v5i12.06

Fuzzy Based Implementation of Multi - Keyword Ranked Search over Encrypted Cloud Data in Secure Cloud Environment

Gladiss Merlin. N

Assistant Professor, Department of Computer Science and Engineering
Jeppiaar Institute of Technology

Abstract
Cloud computing enables the paradigm of data outsourcing. However, to protect data privacy, sensitive cloud data have to be encrypted before outsourced to the commercial public cloud, which makes effective data utilization service is a challenging task. Even though searchable encryption techniques allow users to securely search over encrypted data through keywords, they support only Boolean search and are not yet sufficient to meet the effective data utilization need that is inherently demanded by large number of users and huge amount of data files in cloud. Hence it is necessary to allow multiple keywords in the search request and return documents in the order of their relevance to the keywords. The works on searchable encryption focused on single keyword search or Boolean keyword search and the result produced by them are rarely sorted. An effective method proposed for this challenging problem is privacy-preserving search over encrypted cloud data. This method establishes a set of strict privacy requirements for such a secure cloud data utilization system through MRSE. Among various multi-keyword semantics, this method chooses the efficient similarity measure of "coordinate matching". Then according to Top K Query method the sorted results are produced. The privacy is preserved by the chunk of data stored in a various server's. Then further improvisation is taken to introduce low overhead on computation and communication in future.

Keywords—cloud computing;keyword search; top k query; encryption; MRSE

I. Introduction

Cloud computing has been envisioned as the next generation information technology architecture for enterprises, due to its long list of unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk. As a disruptive technology with profound implications, cloud computing is transforming the very nature of how businesses use information technology [1]. Several trends are opening up the era of cloud computing, which is an Internet-based development and use of computer technology. The ever cheaper and more powerful processors, together with the Software as a Service computing architecture, are transforming data centres into pools of computing service on a huge scale. The increasing network bandwidth and reliable yet flexible network connections make it even possible that users can now subscribe high quality services from data and software that reside solely on remote data centre's [2].

To protect data privacy and combat unsolicited accesses in the cloud and beyond, sensitive data, Cloud service providers (CSP) usually enforce user's data security through mechanisms like firewalls and virtualization. However, these mechanisms do not protect user's privacy from the CSP itself since the CSP possesses full control of the system hardware and lower levels of software stack [4]. Therefore encryption before data outsourcing the cloud data; this, however, obsoletes the traditional data utilization service based on plaintext keyword search. The trivial Solution of downloading all the data and decrypting locally is clearly impractical, due to the huge amount of bandwidth cost in cloud scale systems. Thus, exploring privacy preserving and effective search service over encrypted cloud data is of paramount importance. Considering the potentially large number of on-demand data users and huge amount of outsourced data documents in the cloud, this problem is particularly challenging as it is extremely difficult to meet also the requirements of performance, system usability, and scalability.

On the one hand, to meet the effective data retrieval, the large amount of documents demand the cloud server to perform result relevance ranking, instead of returning undifferentiated results. Such ranked search system enables data users to find the most relevant information quickly, rather than burdensomely sorting through every match in the content collection. On the other hand, to improve the search result accuracy as well as to enhance the user searching experience, it is also necessary for such ranking system to support multiple keywords search, as single keyword search often yields far too coarse results [3]. To provide more accuracy to the end user's result is done by searching, the unlabelled data keyword's are included in the index of the server and then searching is done this search results is then categorized and then they are sorted in their division using Top k query algorithm. TOP-k selection queries will help to sort the relevant data and provide the exact data to the end user.

II. Related Works

A mechanism Searchable Keyword-Based Encryption (SKBE) considers decrypting the searched results as well as searching for desired documents. In addition to searching ability in existing schemes, SKBE's another goal is to enable a user to give a proxy the ability to decrypt only the encryptions containing desired keywords, but not other encryptions. For providing the most powerful functionality, it is designed for supporting conjunctive keyword search in a public key setting. SKBE is a public key encryption with the following functionalities. A message is encrypted using a public key. Then, the cipher text depends on the keywords associated with the message. Given certain information called a decrypt trapdoor for keywords, cipher texts containing all of keywords can be decrypted without a private key. Similar to searchable encryptions, given certain information called a search trapdoor for keywords, we can test whether a cipher text contains keywords all, but get no other information about its original document. These trapdoors can be generated only with. Without a trapdoor, a cipher text does not reveal anything about its corresponding document.

To enrich search functionalities, conjunctive keyword search over encrypted data have been proposed. These schemes incur large overhead caused by their fundamental primitives, such as computation cost by bilinear map, for example, or communication cost by secret sharing. As a more general search approach, predicate encryption schemes are recently proposed to support both conjunctive and disjunctive search. Conjunctive keyword search returns "all-or-nothing," which means it only returns those documents in which all the keywords specified by the search query appear; disjunctive keyword search returns undifferentiated results, which means it returns every document that contains a subset of the specific keywords, even only one keyword of interest. In short, none of existing Boolean keyword searchable encryption schemes support multiple keywords ranked search over encrypted cloud data while preserving privacy as we propose to explore in this paper. Without providing the capability to compare concealed inner products, predicate encryption is not qualified for performing ranked search.

Furthermore, most of these schemes are built upon the expensive evaluation of pairing operations on elliptic curves. Such inefficiency disadvantage also limits their practical performance when deployed in the cloud. Our early work [3] has been aware of this problem, and provides solutions to the multi-keyword ranked search over encrypted data problem. In this paper, we extend and improve more technical details as compared to [3]. We propose two new schemes to support more search semantics which improve the search experience of the MRSE scheme, and also study the dynamic operation on the data set and index which addresses some important yet practical considerations for the MRSE design. On a different front, the research on top-k retrieval in database community is also loosely connected to our problem.

Departing from many previous works that focused on queries consisting of a single keyword, we consider the case of queries consisting of arbitrary Boolean expressions on keywords, that is to say conjunctions and disjunctions of keywords and their complement. Our construction of Boolean symmetric searchable encryption BSSE is mainly based on the orthogonalization of the keyword field according to the Gram- Schmidt process. Each document stored in an outsourced server is associated with a label which contains all the keywords corresponding to the document, and searches are performed by way of a simple inner product. Furthermore, the queries in the BSSE scheme are randomized [3]. This randomization hides the search pattern of the user since the search results cannot be associated deterministically to queries. In addition, the search complexity is in O(n) where n is the number of documents stored in the outsourced server.

A single pair of virtual appliance and unit will not be able to fulfill all the requirements of a business problem. Inevitably most problems will require more than one of those services working together to provide a complete solution. Hence it is important to develop compositions of them. Konstantinou et al. [23] proposed an approach to plan, model, and deploy Cloud service compositions. In their approach, the solution model and the deployment plan for the composition in Cloud platform are developed by skilled users and executed by unskilled users. Likewise, in our system set of compatibility constraints from experts were captured which would be utilized to simplify the process of deployment for end users by eliminating invalid compositions solutions. However, as they also mentioned, their work lacks an approach for appliance selection and their placement on the Cloud which is offered by our work. Similarly Chieu et al. [24] proposed the use of composite appliances to automate the deployment of integrated solutions. However in their work as well, QoS objectives are not considered when building the composition.

Similarly another work has utilized Intuitionistic Fuzzy Set (IFS) for ranking service compositions in the context of Grid and SOA [25]. It does not deal with users' constraints such as compatibility and when the problem is NP-hard (like our service composition problem) the execution time is not acceptable.

There are other studies and toolkits that offered ontology modeling for Cloud services. Unified Cloud Interface (UCI) provides ontology4 model for modeling Amazon EC2 services. Mosaic project [26] is proposed to develop multi-Cloud oriented applications. In Mosaic, Cloud ontology plays and essential role, and expresses the application's needs for Cloud resources in terms of SLAs and QoS requirements. It is utilized to offer a common access to Cloud services in Cloud federations. However, none of these ontologies focus on modeling of compatibility of Cloud services.

There are several existing approaches [27], [28], [29], [30] that are capable of dealing with incompatible services. However, many of them only focused on compatibility of Input and Output (I/O) of services in a composition. In our case we are not concerned of I/O level incompatibilities; instead we are interested in modeling incompatibilities that are caused by regulations and other factors that are not related to service functionalities and I/O.

While there are a number of other studies that focus virtual machine and appliance selection and deployment problem, we are not aware of any work that provides a framework for composing and deploying multiple virtual appliances with the focus on automatic compatibility checking.

III. MRSE-System Formation

The Efficient MRSE in cloud server is achieved by considering a cloud data hosting service involving three different entities they are the data owner, the data user, and the cloud server.

The data owner has a collection of data documents F to be outsourced to the cloud server inthe encrypted form C. To enhance the searching capability over C for effectivedatautilization, the data owner, will first build an encrypted searchable index I from F, and then outsource both the index I and the encrypted document collection C to the cloud server. The document collection C is referred to both labelled and unlabelled documents.

To search the document collection for t given keywords, an authenticated user requires a corresponding trapdoor T through search control mechanisms

Upon receiving T from a data user, the cloud server is responsible to search the index I and return the corresponding set of encrypted documents. To improve the document retrieval accuracy, the search result should be ranked by the cloud server according to some ranking criteria. And to reduce the communication cost, the data user may send an optional number k along with the trapdoor T so that the cloud server only sends back top-k documents that are most relevant to the search query. To enhance the privacy in cloud server the documents are made as chunk and stored in servers.

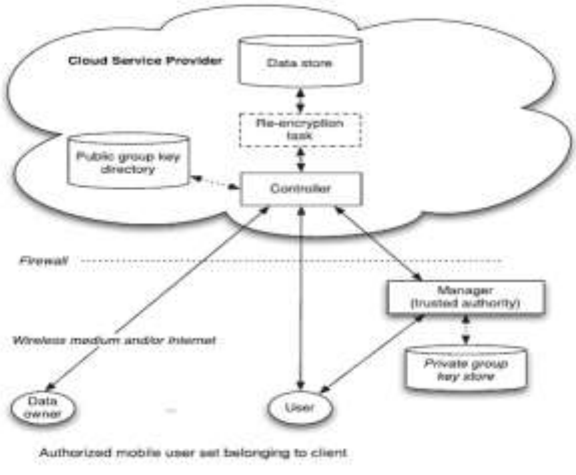

Cloud Service Provider
Data store
Re-encryption task
Public group key directory
Controller
Firewall
Manager (trusted authority)
Wireless medium and/or internet
Private group key store
Data owner
User
Authorized mobile user set belonging to client

Fig1: Overview Approach of MRSE Formation

Finally, the access control mechanism is employed to manage decryption capabilities given to users and the data collection can be updated in terms of inserting new documents, updating existing documents, and deleting existing documents.

IV. Privacy Preserving And Efficient Search On Cloud Computing

Privacy-preserving query over encrypted graph-structured data in cloud computing establish a strict privacy requirements for a secure cloud data utilization system to become practical. This graph structured data utilizes the principle of "filtering-and-verification". This method implements efficient inner product as the pruning tool to carry out the filtering procedure [6].

The major issue is of security in service outsourcing: the elements of an encryption scheme and the execution protocol for encrypted query processing. The model of secure query processing is SCONEDB (Secure Computation ON an Encrypted Database). The conventional way to deal with security threats is to apply encryption on the plain data and to allow only authorized parties to perform decryption. Unauthorized parties, including the service provider, should not be able to recover the plain data even if they can access the encrypted database. Some previous works have experienced the encryption problem in the outsourced database model. However, these studies are restricted to simple SQL operations, e.g., exact match of attribute value in point query; comparisons between numeric values in range query. In practice, users often interact with a database via applications in which queries are not easily expressible in SQL. Moreover, most of the previous methods were specially engineered to work against one specific attack model. However, the problem should consider with respect to various security requirements, considering different attacker capabilities. And then the arrival of K-nearest neighbor (kNN) algorithm enhances the security and also it express the various encryption schemes that are designed to support secure kNN query processing under different attacker possibilities [10].

Supporting effective data search operations over outsourced cloud data in a privacy-preserving manner is another key challenge. Existing techniques such as searchable encryption are either too computationally expensive, or lack the enough flexibility and usability to be adopted by cloud users in practice. So the focus on achieving ranked/similarity search, multi-keyword search, multi-dimensional range query, and graph-structured data, by combining lightweight cryptography primitives with information-retrieval techniques in novel ways. The existing solution is based on the MD algorithm and cosine similarity, and a scalar-product preserving encryption scheme, which achieves better-than-linear search complexity while preserving data and query privacy simultaneously.

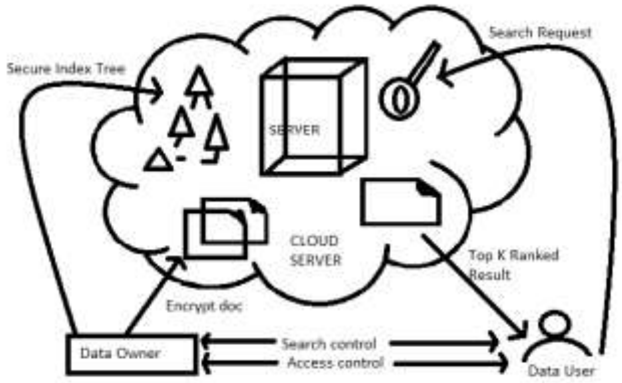


Fig 2 :Privacy-preservingoverclouddata

Ranking search will greatly enhances the system usability by enabling search result relevance ranking instead of sending undifferentiated results. Then the exploration of the Statistical Measure Approach will provide the information retrieval to develop a secure searchable index, and develop a one-to-many order-preserving mapping technique to protect the sensitive score details. The outcome of this design will facilitate the efficient server-side ranking without missing keyword privacy [8].

Advantage of ranking search has taking one step closer towards practical deployment of privacy-preserving service in Cloud Computing. The ideal construction of ranked keyword search under the state-of-the-art searchable symmetric encryption security definition, and demonstrate its inefficiency. To achieve more practical performance the definition for ranked searchable symmetric encryption, that give an efficient design by utilizing the existing cryptographic primitive, order-preserving symmetric encryption .Thorough analysis it shows that solution provides security guarantee compared to previous searchable symmetric encryption schemes, while correctly realizing the goal of ranked keyword search. Extensive experimental results demonstrate the efficiency of the solution among the search [11].

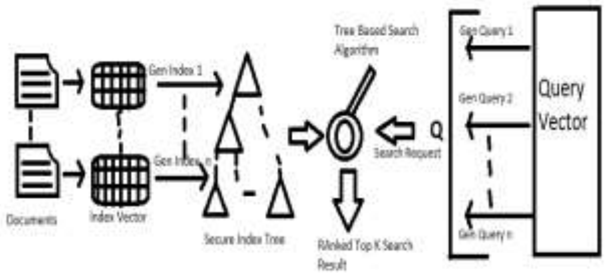


Fig 3: Usable search schemes over cloud data

V. MRSE Framework

In this section, we define the framework of multi-keyword ranked search over encrypted cloud data (MRSE) and establish various strict systemwise privacy requirements for such a secure cloud data utilization system. The operations on the data documents are not shown in the framework since the data owner could easily employ the traditional symmetric key cryptography to encrypt and then outsource data. With focus on the index and query, the MRSE system consists of four algorithms as follows--***Setup* (1^{l}).** Taking a security parameter 'l' as input, the data owner outputs a symmetric key as SK.***BuildIndex* (Ƒ, SK).** Based on the data set Ƒ, the data owner builds a searchable index I which is encrypted by the symmetric key SK and then outsourced to the cloud server. After the index construction, the document collection can be independently encrypted and outsourced. ***Trapdoor* (ῶ).**With t keywords of interest in ῶ as input, this algorithm generates a corresponding trapdoor $T_{\tilde{\omega}}$.***Query($T_{\tilde{\omega}}$,k,I).*** When the cloud server receives a query request as ($T_{\tilde{\omega}}$,k), it performs the ranked search on the index I with the help of trapdoor $T_{\tilde{\omega}}$, and finally returns $\mathcal{F}_{\tilde{\omega}}$, the ranked id list of top-k documents sorted by their similarity with ῶ.

Neither the search control nor the access control is within the scope of this paper. While the former is to regulate how authorized users acquiretrapdoors, the later is to manage users' access to outsourced documents.

- Ƒ—the plaintext document collection, denoted as a set of m data documents Ƒ=(F_1, F_2...F_m).
- C—the encrypted document collection stored in the cloud server, denoted as C=(C_1, C_2...C_m).
- W—the dictionary, i.e., the keyword set consisting of n keyword, denoted as W=(W_1,W_2...W_n).
- I—the searchable index associated with C, denoted as (I_1,I_2...I_m) where each sub index I_i is built for F_i.
- ῶ—the subset of W,representing the keywords in a search request,denoted as ῶ = (W_{j1}, W_{j2}...W_{jn}).
- $T_{\tilde{\omega}}$ —the trapdoor for the search request ῶ
- $\mathcal{F}_{\tilde{\omega}}$ —the ranked id list of all documents according to their relevance to ῶ.

TF–IDF, short for term frequency–inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining. The TF–IDF value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to control for the fact that some words are generally more common than others.

Variations of the TF–IDFweighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query. TF–IDFcan be successfully used for stop-words filtering in various subject fields including text summarization and classification.

$$\mathbf{TF(t,d)}= 0.5 + \frac{0.5 \times f(t,d)}{\max\{f(w,d): w \in d\}}$$

$$\mathbf{IDF(t,D)}= \log \frac{N}{|\{d \in D: t \in d\}|}$$

- N: total number of documents in the corpus
- $|\{d \in D: t \in d\}|$: Number of documents where the term tappears (i.e., tf(t,d)≠0). If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the denominator to 1+$|\{d \in D: t \in d\}|$.

Mathematically the base of the log function does not matter and constitutes a constant multiplicative factor towards the overall result.Then tf–idf is calculated as,

$$\mathbf{TFIDF\ (t,d,D) = tf(t,d)*idf(t,D)}$$

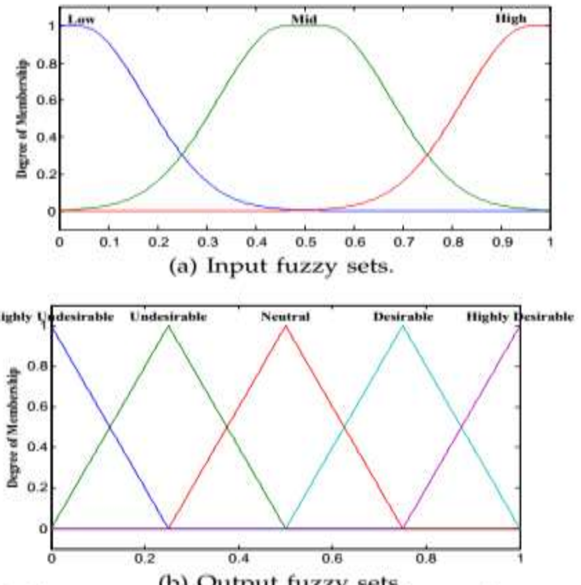
A high weight in tf–idf is reached by a high term frequency and a low document frequency of the term in the whole collection of documents; the weights hence tend to filter out common terms. Since the ratio inside the idf's log function is always greater than or equal to 1, the value is greater than or equal to 0. As term appears in more number of documents, the ratio inside the logarithm approaches 1 and it bring the idf and tf-idf closer to 0.

VI. Results And Discussion

In order to realize and evaluate the proposed approach, a number of components and technologies are utilized. Most importantly, multiobjective algorithms are implemented in jMetal [13]. After that, Pareto Front composition solutions from Jmetal have been passed to our fuzzy-logic based ranking components, which utilizes jFuzzyLogic [20] to define the membership functions and rules.

Table1:Sample high level rules set by user

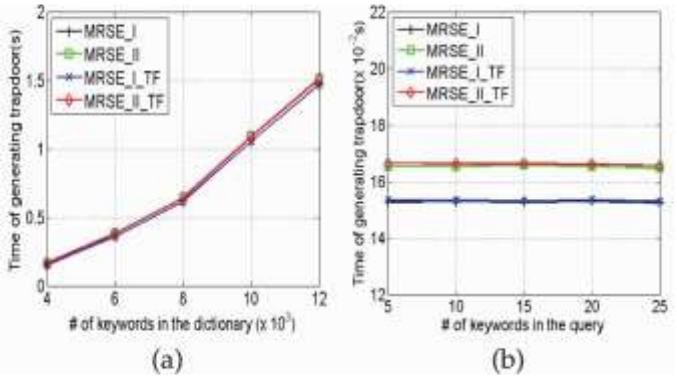
DC	DT	Reliability	Composition Disability
Low	Low	Low	Undesirable
Low	Low	High	Highly Desirable
High	Low	High	Not Sure



(a) Input fuzzy sets.

(b) Output fuzzy sets.

Fig. 4: Fuzzy engine input and output fuzzy sets.



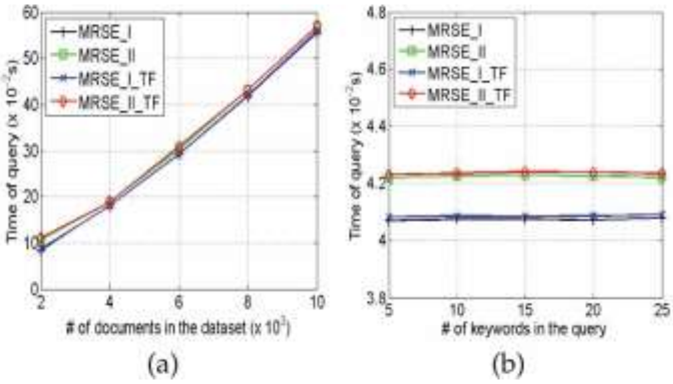


Fig 5: Comparsion of MRSE in diffeerent level

VI. CONCLUSION

In this paper, the solution of multi-keyword ranked search over encrypted cloud data enhances the user to receive the relevant data in the search. And establish a variety of privacy requirements. Here in multi-keyword semantics, the efficient similarity measure of “coordinate matching,” is prosed, that effectively captures the relevance of outsourced documents to the query keywords, and use “inner product similarity” to quantitatively evaluate such similarity measure. For meeting the challenge of supporting multi-keyword semantic without privacy breaches, thebasic idea of MRSE using secure inner product computation is proposed. And to improve the privacy the documents are made into chunk in servers. Using the Term Frequency method the relevancy of the desired document is determined. Along with the Top K Query the determined K documents are produced to the end user. Thorough analysis investigating privacy and efficiency guarantees of proposed schemes is given, and experiments on the real-world data set show our proposed schemes introduce low overhead on both computation and communication.

References

1. C. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing,” Proc. IEEE INFOCOM, 2010.
2. C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, “Toward Secure and Dependable Storage Services in Cloud Computing,” IEEE Trans. Services Computing, vol. 5, no. 2, pp. 220-232, Apr.-June 2012
3. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, “Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data,” Proc. IEEE INFOCOM, , jan, 2014.
4. E.-J. Goh, “Secure Indexes,” Cryptology ePrint Archive, http:// eprint.iacr.org/2003/216. 2003

5. N. Cao, S. Yu, Z. Yang, W. Lou, and Y. Hou, “LT Codes-Based Secure and Reliable Cloud Storage Service,” Proc. IEEE INFO-COM, pp. 693-701, 2012.
6. S. Kamara and K. Lauter, “Cryptographic Cloud Storage,” Proc. 14th Int’l Conf. Financial Cryptograpy and Data Security, Jan. 2010.
7. A. Singhal, “Modern Information Retrieval: A Brief Overview,” IEEE Data Eng. Bull., vol. 24, no. 4, pp. 35-43, Mar. 2001.
8. Y.-C. Chang and M. Mitzenmacher, “Privacy Preserving Keyword Searches on Remote Encrypted Data,” Proc. Third Int’l Conf. Applied Cryptography and Network Security, 2005.
9. R. Curtmola, J.A. Garay, S. Kamara, and R. Ostrovsky, “Searchable Symmetric Encryption: Improved Definitions and Efficient Con- structions,” Proc. 13th ACM Conf. Computer and Comm. Security (CCS ’06), 2006.
10. D. Boneh, G.D. Crescenzo, R. Ostrovsky, and G. Persiano, “Public Key Encryption with Keyword Search,” Proc. Int’l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), 2004.
11. M. Bellare, A. Boldyreva, and A. ONeill, “Deterministic and Efficiently Searchable Encryption,” Proc. 27th Ann. Int’l Cryptology Conf. Advances in Cryptology (CRYPTO ’07), 2007.
12. J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, “Fuzzy Keyword Search Over Encrypted Data in Cloud Computing,” Proc. IEEE INFOCOM, Mar. 2010.
13. D. Boneh, E. Kushilevitz, R. Ostrovsky, and W.E.S. III, “Public Key Encryption That Allows PIR Queries,” Proc. 27th Ann. Int’l Cryptology Conf. Advances in Cryptology (CRYPTO ’07), 2007.
14. P. Golle, J. Staddon, and B. Waters, “Secure Conjunctive Keyword Search over Encrypted Data,” Proc. Applied Cryptography and Network Security, pp. 31-45, 2004.
15. L. Ballard, S. Kamara, and F. Monrose, “Achieving Efficient Conjunctive Keyword Searches over Encrypted Data,” Proc. Seventh Int’l Conf. Information and Comm. Security (ICICS ’05), 2005.
16. Y. Hwang and P. Lee, “Public Key Encryption with Conjunctive Keyword Search and Its Extension to a Multi-User System,” Pairing, vol. 4575, pp. 2-22, 2007.
17. J. Katz, A. Sahai, and B. Waters, “Predicate Encryption Supporting Disjunctions, Polynomial Equations, and Inner Products,” Proc. 27th Ann. Int’l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), 2008.
18. A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, “Fully Secure Functional Encryption: Attribute-Based Encryption and (Hierarchical) Inner Product Encryption,” Proc. 29th Ann. Int’l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT ’10), 2010.
19. E. Shen, E. Shi, and B. Waters, “Predicate Privacy in Encryption Systems,” Proc. Sixth Theory of Cryptography Conf. Theory of Cryptography (TCC), 2009.
20. M. Li, S. Yu, N. Cao, and W. Lou, “Authorized Private Keyword Search over Encrypted Data in Cloud Computing,” Proc. 31st Int’l Conf. Distributed Computing Systems (ICDCS ’10), pp. 383- 392, June 2011.
21. C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, “Secure Ranked Keyword Search over Encrypted Cloud Data,” Proc. IEEE 30th Int’l Conf. Distributed Computing Systems (ICDCS ’10), 2010.
22. C. Wang, N. Cao, K. Ren, and W. Lou, “Enabling Secure and Efficient Ranked Keyword Search over Outsourced CloudData,” IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 8, pp. 1467- 1479, Aug. 2012.
23. A. V. Konstantinou, T. Eilam, M. Kalantar, A. A. Totok, W. Arnold, and E. Snible, “An architecture for virtual solution composition and deployment in infrastructure clouds,” in Proceedings of the 3rd International Workshop on Virtualization Technologies in Distributed Computing, 2009.
24. T. Chieu, A. Mohindra, A. Karve, and A. Segal, “Solutionbased deployment of complex application services on a cloud,” in Proceedings of the 2010 IEEE International Conference on Service Operations and Logistics and Informatics, 2010.
25. P. Wang, “Qos-aware web services selection with intuitionistic fuzzy set under consumer’s vague perception,” Expert Systems with Applications, vol. 36, no. 3, pp. 4460–4466, 2009.

26. B. Di Martino, D. Petcu, R. Cossu, P. Goncalves, T. Máhr, and M. Loichate, “Building a mosaic of clouds,” in Proceedings of Euro-Par 2010 Parallel Processing Workshops. Springer, 2011
27. P. Bartalos and M. Bieliková, “Automatic dynamic web service composition: A survey and problem formalization,” Computing and Informatics, vol. 30, no. 4, pp. 793–827, 2012.
28. F. Rosenberg, M. Muller, P. Leitner, A. Michlmayr, A. Bouguettaya, and S. Dustdar, “Metaheuristic optimization of largescale qos-aware service compositions,” in Proceedings of IEEE International Conference on Services Computing, 2010.
29. F. Lecue and N. Mehandjiev, “Towards scalability of quality driven semantic web service composition,” in Proceedings of IEEE International Conference on Web Services. IEEE, 2009.
30. M. Alrifai, T. Risse, P. Dolog, and W. Nejdl, “A scalable approach for qos-based web service selection,” in Proceedings of Service-Oriented Computing–ICSOC Workshops, 2009.