International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 25 Issue 4 – APRIL 2018.

DEVELOPING AN EFFICIENT KEYWORD SEARCH MODEL FOR CIPHERED CLOUD DATA

K. Mahalakshmi[#1] and A. Lavanya[*2]

[#]*M.Sc (Computer science), Department of Computer science, Kamban College of Arts and Science for women, India.*

[*] *Head of the Department, Department of Computer science, Kamban College of Arts and Science for women, India.*

Abstract: Cloud computing is a new inspiration technology which efficiently support the client oriented services. Now in these days, there are figures of applications which devour the cloud storage service for keep and get back information. In such state, the data owner management and privacy preservation cryptographic techniques are make use of frequently. In this paper, we propose an efficient keyword search model for the ciphered cloud data. Access Control is the major part of the outsourced database systems. With the help of hierarchical predicate encryption model, the public data is ciphered. The cloud data enforces fine-grained access control and perform multi-field query search operation systems. The main objective of this system is to enhance the searching operation of the cloud data which help to achieve searching capability and keywords update. Experimental results have shown the efficiency of the proposed systems.

Keywords: Cloud computing, Access Control, Keyword search, hierarchical predicate encryption and Searching complexity.

I. INTRODUCTION

Cloud storage outsourcing has become a popular application for enterprises and organizations to reduce the burden of maintaining big data in recent years. However, in reality end users may not entirely trust the cloud storage servers and may prefer to encrypt their data before uploading them to the cloud server in order to protect the data privacy. This usually makes the data utilization more difficult than the traditional storage where data is kept in the absence of encryption. One of the typical solutions is the searchable encryption which allows the user to retrieve the encrypted documents that contain the user-specified keywords, where given the keyword trapdoor, the server can find the data required by the user without decryption. The PHRs are encrypted in order to comply with privacy regulations like HIPAA. Note that the context we are considering supports private data sharing among multiple data providers and multiple data users. Therefore, SE schemes in the private key setting [1], which assume that a single user who searches and retrieves his/her own data are not suitable. In a PEKS system, using the receiver's public key, the sender attaches some encrypted keywords (referred to as PEKS ciphertexts) with the encrypted data. The receiver then sends the trapdoor of a to-be-searched keyword to the server for data searching.

Therefore, SE schemes in the private key setting [2], which assume that single users who searches and retrieves his/her own data are not suitable. In a PEKS system, using the receiver's public key, the sender attaches some encrypted keywords (referred to as PEKS ciphertexts) with the encrypted data. The receiver then sends the trapdoor of a to-be-searched keyword to the server for data searching. Given the trapdoor and the PEKS ciphertext, the server can test whether the keyword underlying the PEKS ciphertext is equal to the one selected by the receiver. If so, the server sends the matching encrypted data to the receiver. PEKS schemes suffer from an inherent insecurity regarding the trapdoor keyword privacy, namely inside Keyword Guessing Attack (KGA). The reason leading to such security vulnerability is that anyone who knows receiver's public key can generate the PEKS ciphertext of arbitrary keyword himself. Specifically, given a trapdoor, the adversarial server can choose a guessing keyword from the keyword space and then use the keyword to generate a PEKS ciphertext. The server then can test whether the guessing keyword is the one underlying the trapdoor. This guessing-then-testing procedure can be repeated until the correct keyword is found. Such a guessing attack has also been considered in many password-based systems.

Let us assume that the parties are semi-honest and do not collude with each other to bypass the security measures. In an offline stage, the data owner creates a search index for each document. The search index file is created using a secret key based trapdoor generation function where the secret keys1 are only known by the data owner. Then, the data owner uploads these search index files to the server together with the encrypted documents. We use symmetric-key encryption as the encryption method since it can handle large document sizes efficiently [3]. This process is referred as the index generation henceforth and the trapdoor generation is considered as its one of the steps.

The rest of the paper is organized as follows: Section II presents the related work; Section III presents the proposed work; Section IV presents the experimental analysis and finally concludes in Section V.

II. RELATED WORK

This section presents the prior techniques established in keyword search model on cloud data.

The author in [4] MTS scheme with similarity-based ranking was studied. Author builds the search index based on term frequency and the vector space model with cosine similarity measure to attain higher search result accuracy. To enhance the search efficiency, a tree-based index structure for multi-dimensional (MD) algorithm are proposed so that the search efficiency is much better than that of linear search, also two secure index schemes to satisfy the stringent privacy requirements under strong threat models, i.e., known ciphertext model and known background model. It provides the better efficiency but result in precision loss. Several methods suggested which simultaneously supports dynamic update operations of documents. Specifically, the vector space model and the widely-used TF*IDF model are combined within the index construction and query generation. For efficient multi-keyword rank search here proposed a special tree-based index structure and named it a Greedy Depth-first Search algorithm. Due to the special structure of our tree-based index, the proposed scheme can flexibly achieve sub linear search time and cope with the deletion and insertion of documents.

The author in [5] presented the approximate String Search in Spatial Databases using MHR tree concept. It is based on R-Tree augmented with min wise signature and linear hashing. Keyword are going to be search using hash keys which are identical to related set and element. Pruning the tree according to signature and query string. During this tree information is arranged in tree form before giving any query and after we give any query then we get hash code associate to keyword. So, it is more efficient, low time cost required compare to alternative [6]. The multidimensional binary search tree is a data structure for storage of data to be retrieved by associative searches. A major advantage of this structure is that a single data structure can handle many varieties of queries very efficiently. To provide a security to data there is need of encryption of data [7].

RSA involves two keys a public key and a private key. Anyone will use the public key to encrypt a message, but only someone with knowledge of the nonpublic (private) key can hope to decrypt the message in a reasonable amount of time. The full decryption of an RSA cipher text is unfeasible because no efficient rule currently exists for factoring large numbers. AES [8] is based combination of both substitution and permutation, and is fast in both software and hardware. AES could be a variant of Rijndael that has a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits. AES provide the very excellent security over RSA. Later around 2004 the concept of KP-ABE was introduced. John Bethencourt [9] explicit attribute base encryption. Attribute-based encryption [10] could be a sort of public-key encryption in which the secret key of a user and the cipher text are dependent upon attributes (e.g. the country he lives, or the type of subscription he has). In such a system, the decryption of a cipher text is feasible only if the set of attributes of the user key matches the attributes of the cipher text.

III. PROPOSED WORK

This section presents the proposed work of our research study. The main objectives of this study are:

- To propose a scalable framework that integrates multi-field keyword search with fine-grained access control.
- To make a novel use of Hierarchical Predicate Encryption (HPE), to realize the derivation of search capability.

The proposed keyword search model composes of four phases, namely,

A) Cloud Users (CU):

User's stores a great quantity of data files in the cloud can be an individual or a organization. Cloud users (data owners), who outsource their Encrypted data in clouds. Users can be relieved of the burden of storage and computation while enjoying the storage and maintenance service by outsourcing their data into the CSP.

B) Cloud Service Providers (CSP):

A cloud service provider is a third-party company offering a cloud-based platform, infrastructure, and application or storage services. Much like a homeowner would pay for a utility such as electricity or gas; companies typically have to pay only for the amount of cloud services they use, as business demands require. Besides the pay-per-use model, cloud service providers also give companies a wide range of benefits. Businesses can take advantage of scalability and flexibility by not being limited to physical constraints of on-premises servers, the reliability of multiple data centers with multiple redundancies, customization by configuring servers to your preferences and responsive load balancing which can easily respond to changing demands. Though businesses should also evaluate security considerations of storing information in the cloud to ensure industry-recommended access and compliance management configurations and practices are enacted and met. Cloud Service Provider Manages and coordinates a number of cloud servers to offer scalable and on-demand outsourcing data services for users.

C) Third Party Auditors (TPA):

The reliability of the cloud storage system is achieved by the Third Party Auditors (TPA). It works on the basis of request received from the users. TPA is involved to check the integrity of the users data stored in the cloud. However, in the whole verification process, the TPA is not expected to be able to learn the actual content of the user's data for privacy protection. It is also assumed that TPA is credible but curious. In otherwords, the TPA can perform the audit reliably, but may be curious about the user's data.

D) Dynamic Hash Table (DHT):

A hash table is a dynamic set data structure. It has three basic functions: to store data (SET/INSERT); to retrieve data (SEARCH/RETRIEVE), and to remove data that has previously been stored in the set (DELETE). In this way it is not different from other dynamic set data structure such as linked lists or trees. The interesting about hash tables is their performance characteristics with respect to the store/retrieve/remove operations. In this regard, hash tables offer average constant time to perform any combination of the basic operations. This makes them extremely useful in many scenarios where quickly searching for an element is required, especially if multiple queries must be performed.

By this proposed model, the following merits achieved are:

- This design leverages the computation power of cloud server.
- It also solves the second challenge by dispersing the computation burden of capability generation to the users in the system.
- It enables the service of both the keyword search and access control over multiple fields, and supports efficient update of access policy and keywords. KSAC also introduces some random values to enhance the protection of user's access privacy.

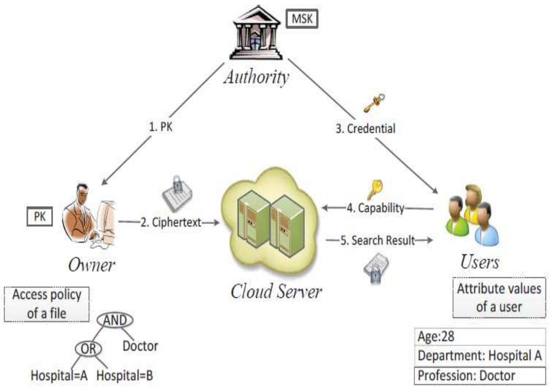


Fig.3.1 System architecture

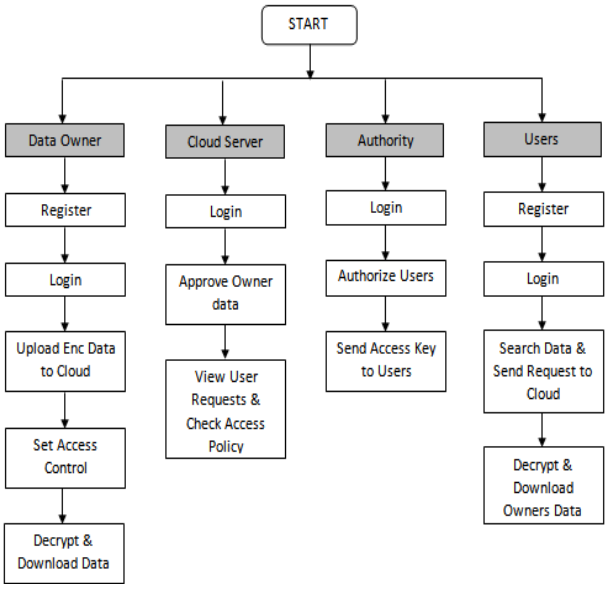
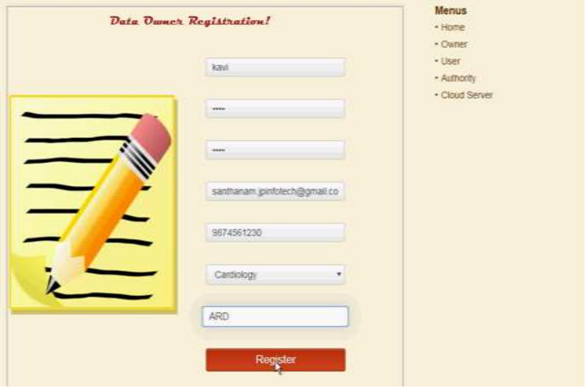


Fig.3.2 Proposed workflow

IV. EXPERIMENTAL RESULTS AND ANALYSIS

This section explores the experimental analysis of the proposed model in DOT NET framework.


Data Owner Registration!

kavi

----

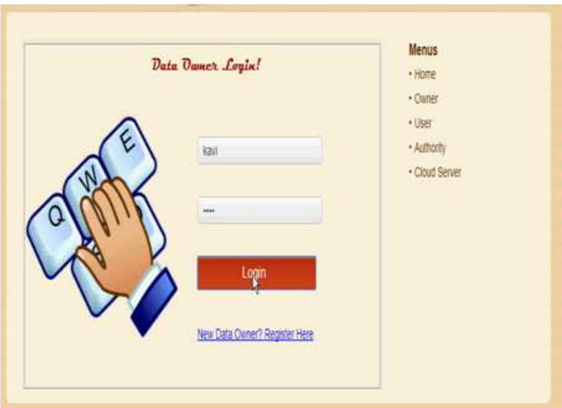
santhanam.jpinfotech@gmail.co

9874561230

Cardiology

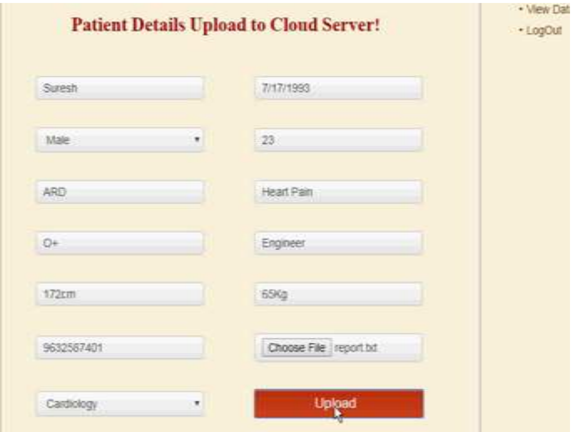
ARD

Fig.4.1 Registration of data owner


Data Owner Login!

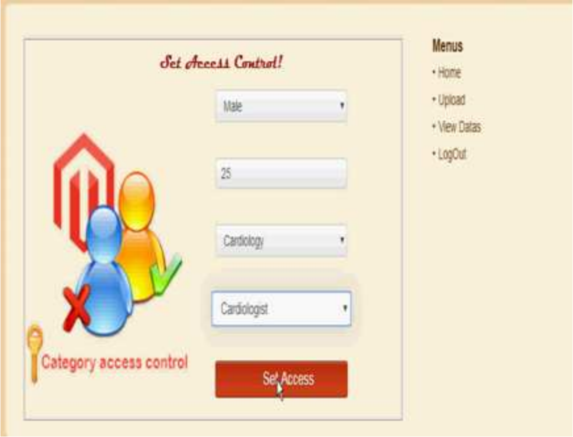
kavi

Fig.4.2 Login process by data owner


Patient Details Upload to Cloud Server!

Suresh	7/17/1993
Male	23
ARD	Heart Pain
O+	Engineer
172cm	65Kg
9632587401	Choose File report.txt
Cardiology	

Fig.4.3 Data owner uploading data to cloud server


Set Access Control!

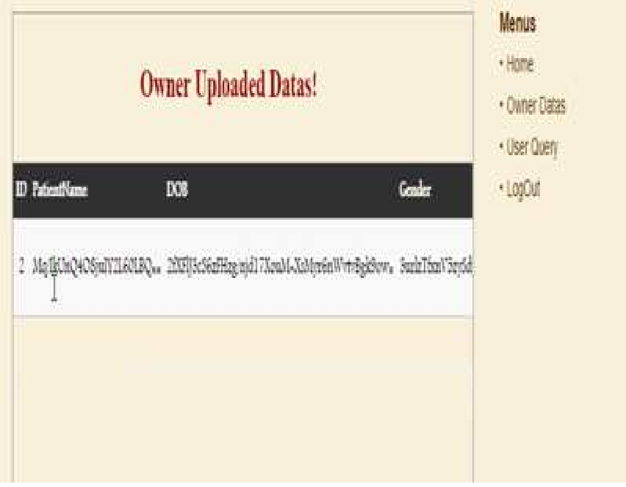
Male

25

Cardiology

Cardiologist

Fig.4.5 Setting access control by data owner


Owner Uploaded Datas!

ID	PatientName	DOB	Gender
2	[illegible]	[illegible]	[illegible]

Fig.4.8 Viewing the uploaded files

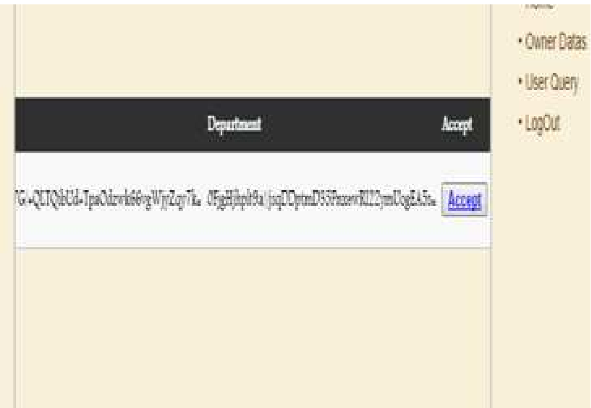
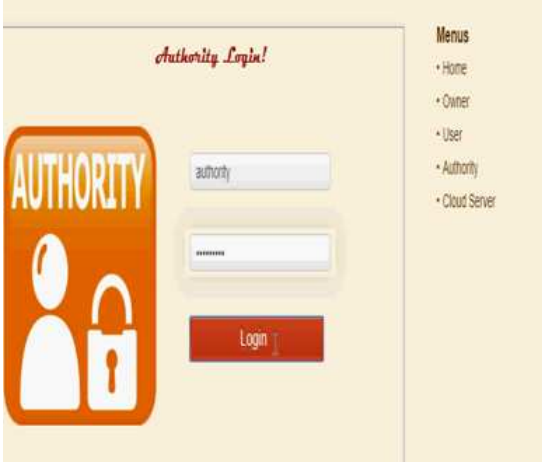

	Department	Accept
[illegible]	[illegible]	Accept

Fig.4.9 Data is accepted by the authorized users

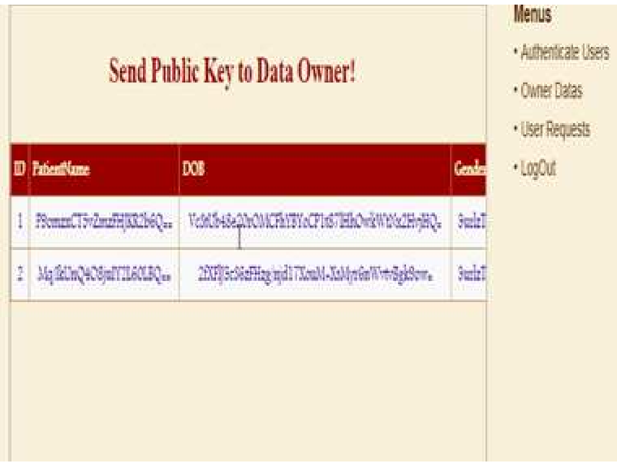


Authority Login!

authority

••••••••

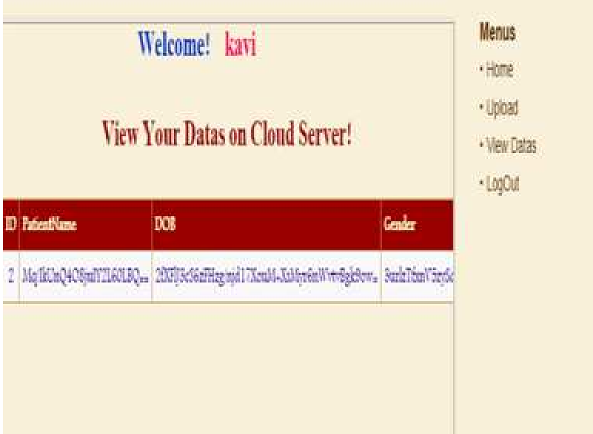
Fig.4.10 Authority’s login



Send Public Key to Data Owner!

ID	PatientName	DOB	Gende
1	F8ommnCT5v2mzFHjKK2b6Q==	Vc6d5b48e2QbOMCFhYBYoCF1x67HHhOwkWtNx2HvjHQ=	3mlzT
2	MqIkUnQ4O8jmlY2L6OLBQ==	2lXFfj5c56zFHzg/njd17XcnM+XaMyr6nWv+vBgk8ow=	3mlzT

Fig.4.11 Sending public key to the data owner

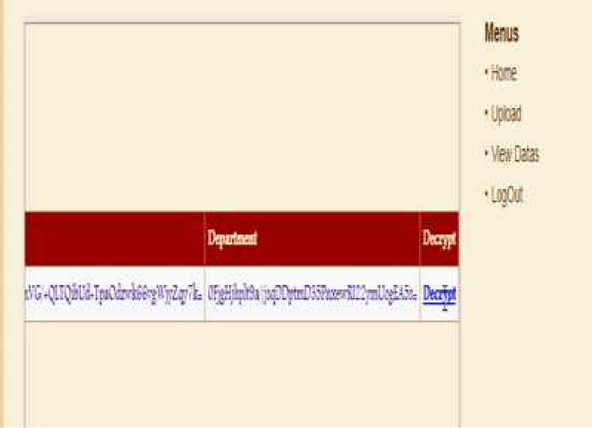


Welcome! kavi

View Your Datas on Clond Server!

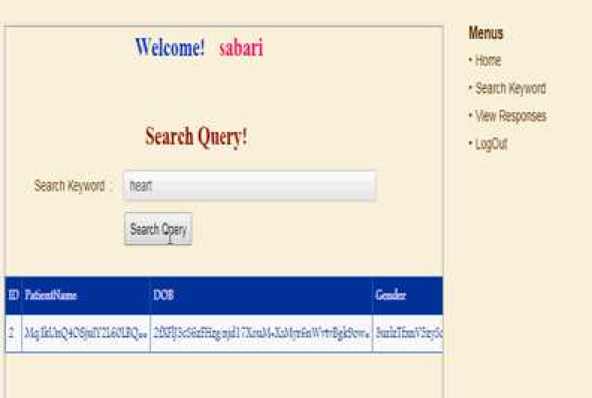
ID	PatientName	DOB	Gender
2	MqIkUnQ4O8jmlY2L6OLBQ==	2lXFfj5c56zFHzg/njd17XcnM+XaMyr6nWv+vBgk8ow=	3mlzTbmV5ry6c

Fig.4.12. Data viewing on cloud server by the cloud users



	Department	Decrypt
rVG+QLTQbUd-Tpa0dzwk66vgWyZqy7k=	UFjgHjbpl9a/jsqDDytmD35PaxewRI22ymU0gEA5o=	Decrypt

Fig.4.13. Data decryption process



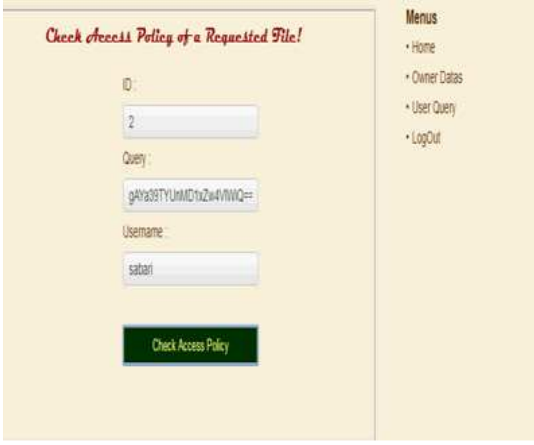
Welcome! sabari

Search Query!

Search Keyword : heart

ID	PatientName	DOB	Gender
2	MqIkUnQ4O8jmlY2L6OLBQ==	2lXFfj5c56zFHzg/njd17XcnM+XaMyr6nWv+vBgk8ow=	3mlzTbmV5ry6c

Fig.4.14. Performing searching operations by authorized users



Check Access Policy of a Requested File!

ID :
2

Query :
gAYa39TYUnMD1xZw4VlWQ==

Username :
sabari

Fig.4.15. Performing data operations on access settings

V. CONCLUSION

We propose secure cloud storage using access control with anonymous authentication. The files are associated with file access policies, that used to access the files placed on the cloud. Uploading and downloading of a file to a cloud with standard Encryption/Decryption is more secure. Revocation is the important scheme that should remove the files of revoked policies. So no one can access the revoked file in future based on attribute revocation scheme. The renew key is added to the file. Whenever the user wants to renew the files he/she may directly download all renew keys and made changes to that keys, then upload the new renew keys to the files stored in the cloud. Although the use of cloud computing has rapidly increased, the security in cloud is major issue, and at the same time users don’t want to lose their data. Experimental analysis has shown the efficiency of the proposed systems.

REFERENCES

[1] Zhirong Shen et al, “Keyword Search with Access Control over Encrypted Cloud Data”, IEEE sensors journal, 2017.

[2] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, “A survey of mobile cloud computing: Architecture, applications, and approaches," Wireless Commun. Mobile Comput., vol. 13, no. 18, pp. 1587_1611, Dec. 2013.

[3] L. Gonzalez-Manzano, A. I. GonzalezTablas, J. M. de Fuentes, and A. Ribagorda, Extended U+F social network protocol: Interoperability, reusability, data protection and indirect relationships in web based social networks, The Journal of Systems and Software, vol. 94, pp. 50–71, 2014.

[4] J. Li and N. Li, Policy-hiding access control in open environment, in Proceedings of the Twenty-Fourth Annual ACM Symposium on Principles of Distributed Computing, ACM, New York, NY, USA, 2005, pp. 29–38.

[5] H. Li, Y. Dai, L. Tian, and H. Yang, ``Identity-based authentication for cloud computing," in Cloud Computing. Berlin, Germany: Springer-Verlag, 2009, pp. 157_166.

[6] H. Liang, L. X. Cai, D. Huang, X. Shen, and D. Peng, “An SMDP-based service model for interdomain resource allocation in mobile cloud networks," IEEE Trans. Veh. Technol., vol. 61, no. 5, pp. 2222_2232, Jun. 2012.

[7] M. M. E. A. Mahmoud and X. Shen, “A cloud-based scheme for protecting sourcelocation privacy against hotspot-locating attack in wireless sensor networks," IEEE Trans. Parallel Distrib. Syst., vol. 23, no. 10, pp. 1805_1818, Oct. 2012

[8] M. Nabeel and E. Bertino, Privacy preserving delegated access control in public clouds, IEEE Trans. Knowl. Data Eng., vol. 26, no. 9, pp. 2268–2280, 2014.

[9] I. Ray, I. Ray, and N. Narasimhamurthi, A cryptographic solution to implement access control in a hierarchy and more, in Proceedings of the Seventh ACM Symposium on Access Control Models and Technologies, ACM, New York, NY, USA, 2002, pp. 65–73.

[10] R. L. Rivest, A. Shamir, and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, Commun. ACM, vol. 21, no. 2, pp. 120–126, 1978.54 Tsinghua Science and Technology, February 2016, 21(1): 40-54.