(IJAER) 2016, Vol. No. 11, Issue No. I, January **e-ISSN: 2231-5152/ p-ISSN: 2454-1796**

SECURE SEARCH SCHEME OF ENCRYPTED DATA ON MOBILE CLOUD

***DEEPA. P , **GOMATHI. S, ***DR. D.C. JOYWINNIEWISE**

** M.E CSE, Francis Xavier Engineering college*
Tirunelveli , India
*** Assistant Professor ,CSE Department*
Francis Xavier Engineering College, Tirunelveli , India
**** HOD & Professor, CSE Department*
Francis Xavier Engineering College, Tirunelveli , India

ABSTRACT

Mobile Cloud Storage acts as the primary file storage for the mobile devices. Data privacy is a very important concern in such cloud storage devices because the mobile device users to store and retrieve files or data on the cloud through wireless communication, which improves the data availability and facilitates the file sharing process without draining the local mobile device resources. In this, a traffic and energy saving encrypted search scheme is used for simplified search and retrieval process that reduces the network traffic for the communication of the selected index and reduces the file retrieval time. This scheme involves three processes such as the process of authentication is used by the data owner to authenticate the data users. The file set and its index are stored in the cloud after being encrypted by the data owner during the preprocessing and indexing stages. The data user searches the files corresponding to a keyword by sending a request to the cloud server in the search and retrieval processes.

I. INTRODUCTION

Cloud computing or in simpler shorthand just "the cloud", also focuses on maximizing the effectiveness of the shared resources. Cloud resources are usually not only shared by multiple users but are also dynamically reallocated per demand. This can work for allocating resources to users. For example, a cloud computer facility that serves European users during European business hours with a specific application (e.g., email) may reallocate the same resources to serve North American users during North America's business hours with a different application (e.g., a web server). This approach should maximize the use of computing power thus reducing environmental damage as well since less power, air conditioning, rack space, etc. are required for a variety of functions. Mobile Cloud Computing (MCC) is the combination of cloud computing, mobile computing and wireless networks to bring rich computational resources to mobile users, network operators, as well as cloud computing providers. The ultimate goal of MCC is to enable execution of rich mobile applications on a plethora of mobile devices, with a rich user experience. MCC provides business opportunities for mobile network operators as well as cloud providers. More comprehensively, MCC can be defined as "a rich mobile computing technology that leverages unified elastic resources of varied clouds and network technologies toward unrestricted functionality, storage, and mobility to serve a multitude of

mobile devices anywhere, anytime through the channel of Ethernet or Internet regardless of heterogeneous environments and platforms based on the pay-as-you-use principle. In this, a traffic and energy saving encrypted search scheme is used for simplified search and retrieval process that reduces the network traffic for the communication of the selected index and reduces the file retrieval time. This scheme involves three processes such as the process of authentication is used by the data owner to authenticate the data users. The file set and its index are stored in the cloud after being encrypted by the data owner during the preprocessing and indexing stages. The data user searches the files corresponding to a keyword by sending a request to the cloud server in the search and retrieval processes

II. RELATED WORKS

Ankur Verma et al. [1] has presents a similarity search include collection of huge data items according to their features, a query that specifics the value of the particular feature and measures the applicability between the query and the data items. In this cloud computing takes encrypted data and performs all the processes to meets the user query without aware of its data, and retrieved encrypted data can be decrypted only by the authorized user who acquaint the request. Antti et al. [2] has provides an analysis of the critical factors affecting the energy consumption of mobile clients in cloud computing. The measurements about the central characteristics of contemporary mobile handheld devices that define the basic balance between local and remote computing are presented further. Bing Wang et al. [3] have proposed a multi keyword fuzzy search scheme by exploiting the locality-sensitive hashing technique. Our proposed scheme achieves fuzzy matching through algorithmic design rather than expanding the index file. It also eliminates the need of a predefined dictionary and effectively supports multiple keyword fuzzy search without increasing the index or search complexity. Cong Wang et al [4] have proposed a secure ranked keyword search over encrypted cloud data. Ranked search greatly enhances system usability by enabling search result relevance ranking instead of sending undifferentiated results, and further ensures the file retrieval accuracy. Cong Wang et al. [5] has proposed ranked search greatly enhances system usability by returning the matching files in a ranked order regarding to certain relevance criteria (e.g., keyword frequency), thus making one step closer towards practical deployment of privacy-preserving data hosting services in Cloud Computing. Jianfeng Wang et al., [6] has proposed new verifiable fuzzy keyword search scheme based on the symbol-tree which not only supports the fuzzy keyword search, but also enjoys the verifiability of the searching result.

III. PROPOSED SYSTEM

To effectively support an encrypted search scheme with a high security level over cloud data, we introduce a new architecture that is named as traffic and energy efficient encrypted keyword search . The basic idea behind traffic and energy efficient encrypted keyword search is to offload the calculation and the ranking load of the relevance scores to the cloud. It has been highlighted that offloading some computation intensive applications onto the cloud can be an efficient low power design philosophy. Cloud providers can provide computing cycles, and users can use these cycles to

reduce the amounts of computation on mobile systems and save energy. However, at the same time, offloaded applications intend to increase the transmission amount and thus increase the energy consumption from another aspect. This double effects motivates us to carefully redesign the traditional file encrypted search and retrieval process. We first take an overview of major processes for all file encrypted search and retrieval schemes. There are normally three main processes:

- The process of authentication is used by the data owner to authenticate the data users.
- The file set and its index are stored in the cloud after being encrypted by the data owner during the preprocessing and indexing stages.
- The data user searches the files corresponding to a keyword by sending a request to the cloud server in the search and retrieval processes.

A. Modified Process of Search and Retrieval

During the preprocessing and indexing stages, the data owner gets a TF table as index and uses Order Preserving Encryption (OPE) to encrypt it. As a result, the cloud server is able to calculate the relevance scores and rank them without decrypting the index. This renders the offloading of the computational load secure and possible.

i) If a data user wants to retrieve the top-k relevant files based on a keyword, he first obtains authentication from the data owner and then receives the keys to encrypt the keyword.

ii) The data user stems the keyword to be queried and encrypts it using the keys.

iii) The data user wraps the encrypted keyword into a tuple, adding some noise to avoid statistic information leak; this tuple is used to perform the retrieval. Then, it is sent to the cloud server together with the number k. The wrap method renders the keywords indistinguishable for an attacker, which will be introduced.

iv) On receiving the wrapped keyword, the cloud server first makes sure that it is accessed by a legal user. If the server is notified by the data owner that this user is to become invalid in a near future, the search is performed but a warning is also issued. If this is a legal user, the server unwraps the tuple to recover the entry of the keyword and searches for it in the index. After calculating the relevance scores, the position of the files corresponding to the keyword is picked and the top k relevant files are sent back to the data user's mobile clients without performing any decryption on these files.

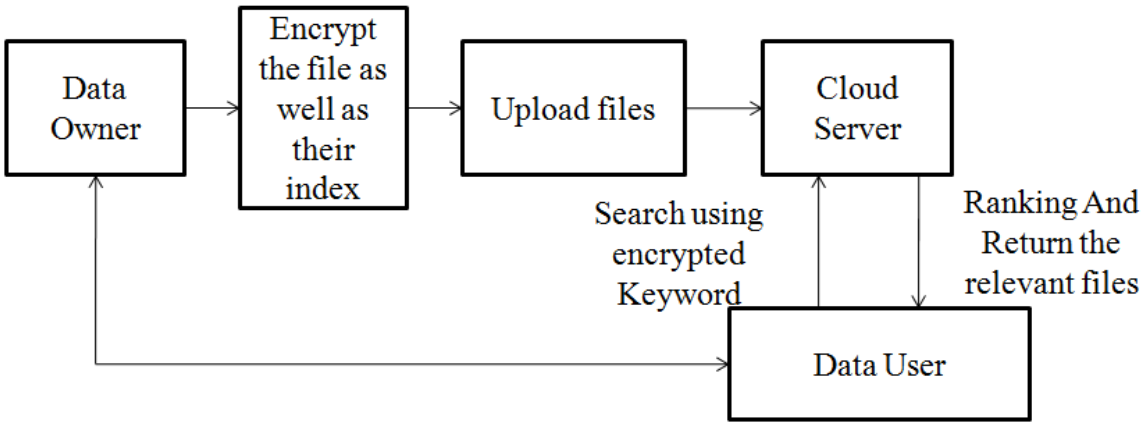

Fig 1 Encrypted Search architecture for traffic and energy efficient encrypted keyword search

v) The data user decrypts these files in the mobile client and recovers the original data. Comparing Fig 3.3 and Fig 3.4, we conclude that the search and retrieval processes in traffic and energy efficient encrypted keyword search are indeed simplified to a single access than TRS. We call it ORS (One Round trip Search), which offload the computation load of "relevance score calculation" from mobile users to the cloud and can intuitively reduce the communication process between the users and cloud server. Moreover, since the relevance score calculation is offloaded to the cloud server, it directly sends the top-k relevant files back to the data user after it receives the retrieval request, which can also reduce the traffic amount for file retrievals at the same time.

B. Onion Routing

Onion routing is a technique for anonymous communication over a computer network. In an onion network, messages are encapsulated in layers of encryption, analogous to layers of an onion. The encrypted data is transmitted through a series of network nodes called onion routers, each of which "peels" away a single layer, uncovering the data's next destination. When the final layer is decrypted, the message arrives at its destination. The sender remains anonymous because each intermediary knows only the location of the immediately preceding and following nodes. An onion is the data structure formed by "wrapping" a message with successive layers of encryption to be decrypted ("peeled" or "unwrapped") by as many intermediary computers as there are layers before arriving at its destination. The original message remains hidden as it is transferred from one node to the next, and no intermediary knows both the origin and final destination of the data, allowing the sender to remain anonymous.

C. Modules

- **Data Owner Module**

The data owner should build a TF table as index and encrypt it using OPE in order to offload the calculation and ranking load of the relevance scores to the cloud. So as to control the statistics information leak, we implement our one-to-many OPE in the data owner module. The authentication between the data owner and the data user is provided in order to ensure the security of traffic and energy efficient encrypted keyword search . In the case of the term frequency (TF) tf (t, d), the simplest choice is to use the raw frequency of a term in a document, i.e. the number of times that term t occurs in document d. If we denote the raw frequency of t by f(t, d), then the simple TF scheme is tf (t, d) = f(t, d).

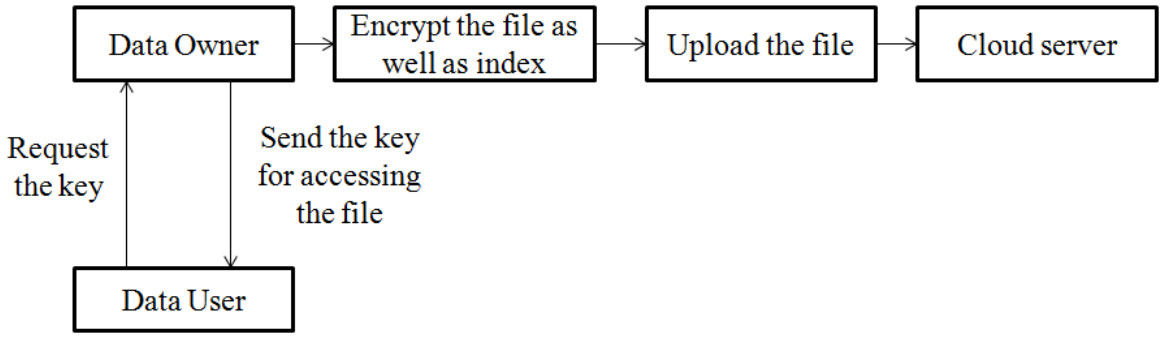

Fig 2 Data flow diagram for data owner module

- **Data User Module**

The data user sends his identity to the data owner and gets the secret keys if authenticated. An authenticated user stems the keyword to be queried, encrypts it with the keys and hashes it to get its entry in the index. Then the encrypted keyword is sent to the cloud server. On receiving the encrypted keyword, the cloud server will find the top-k relevant files and sent back to the data user where the top-k is configured by the users. The data user decrypts the files and recovers the original data.

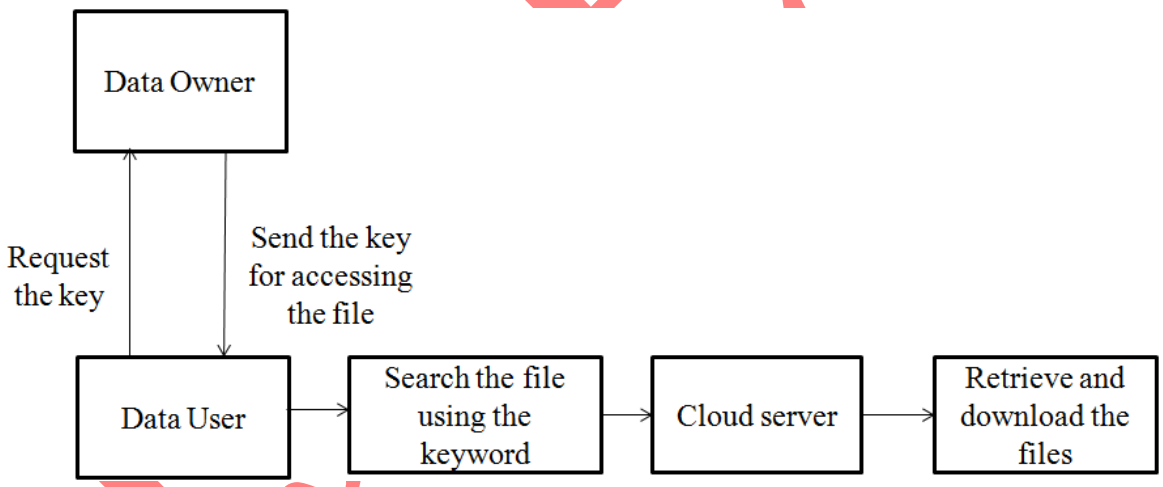

Fig 3 Data flow diagram for data owner module

- **Cloud Server Module**

During the file retrieval process, the authenticated data user sends the encrypted keywords to the cloud server and gets top-k ranked files back.

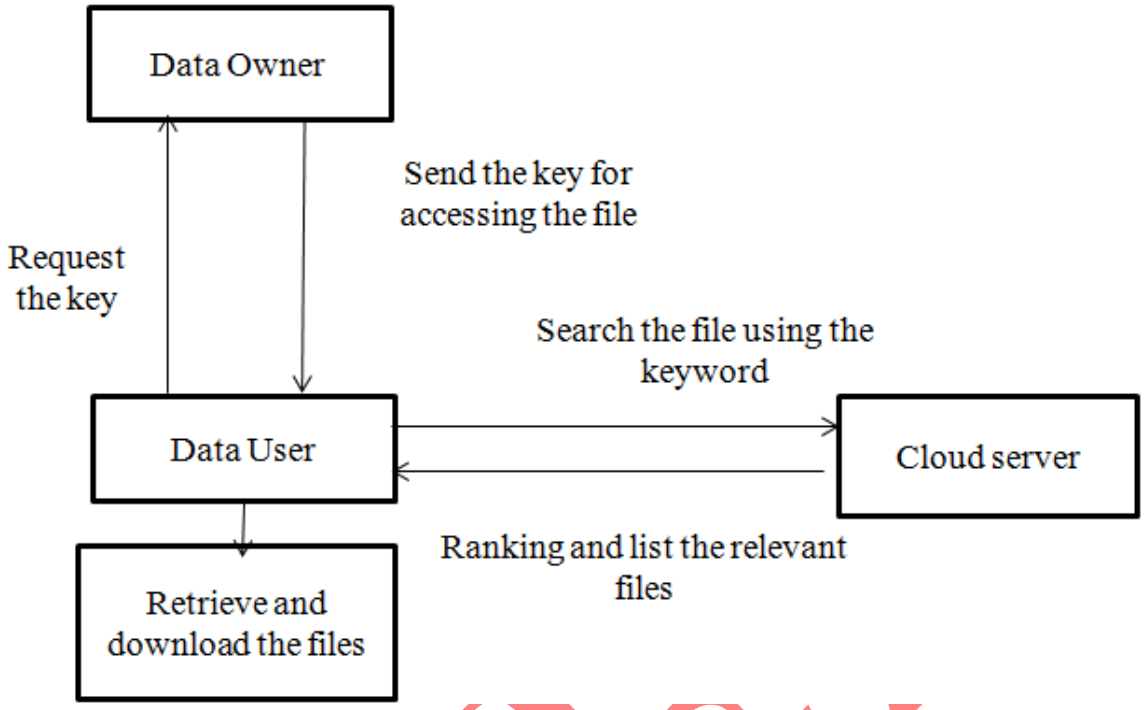

Fig 4 Data flow diagram for cloud server

IV SIMULATION RESULTS

The implementation of this project is done using JAVA. JAVA is an object oriented programming language which derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Programs written in Java are executed at a greater speed.

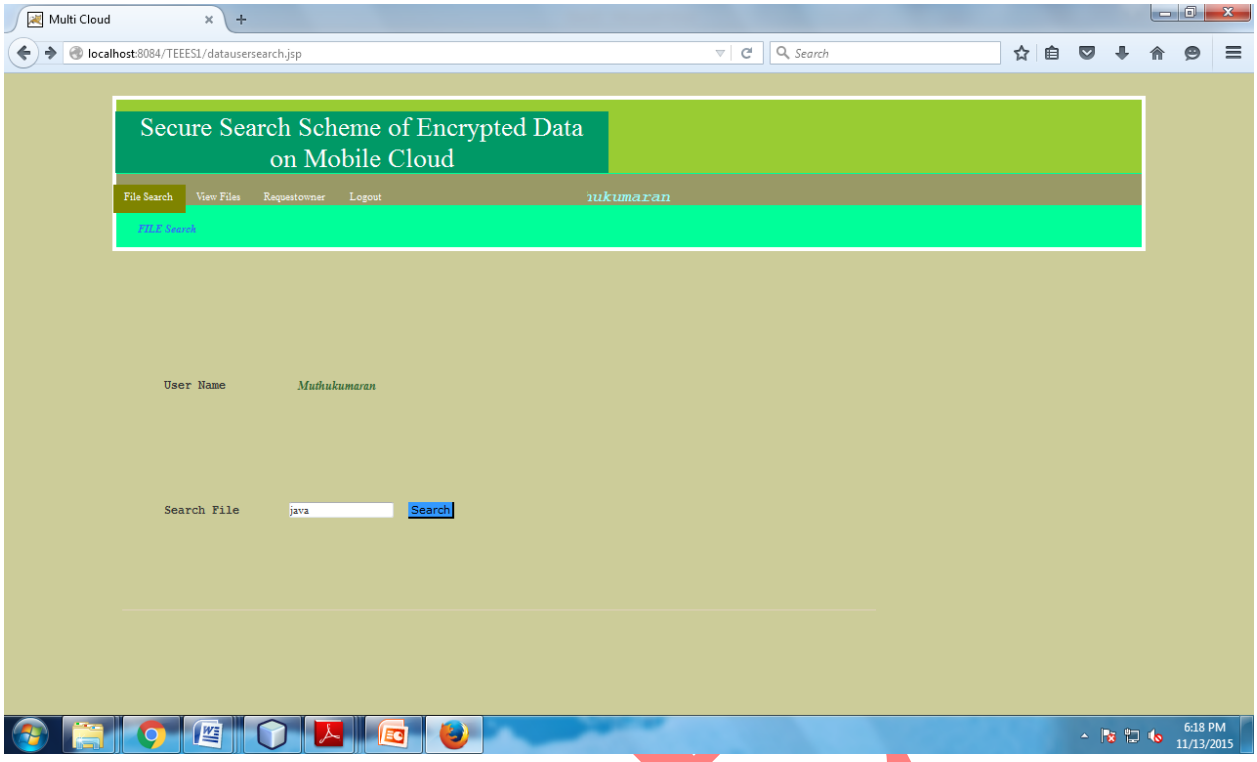

Fig 5 Data user search a file

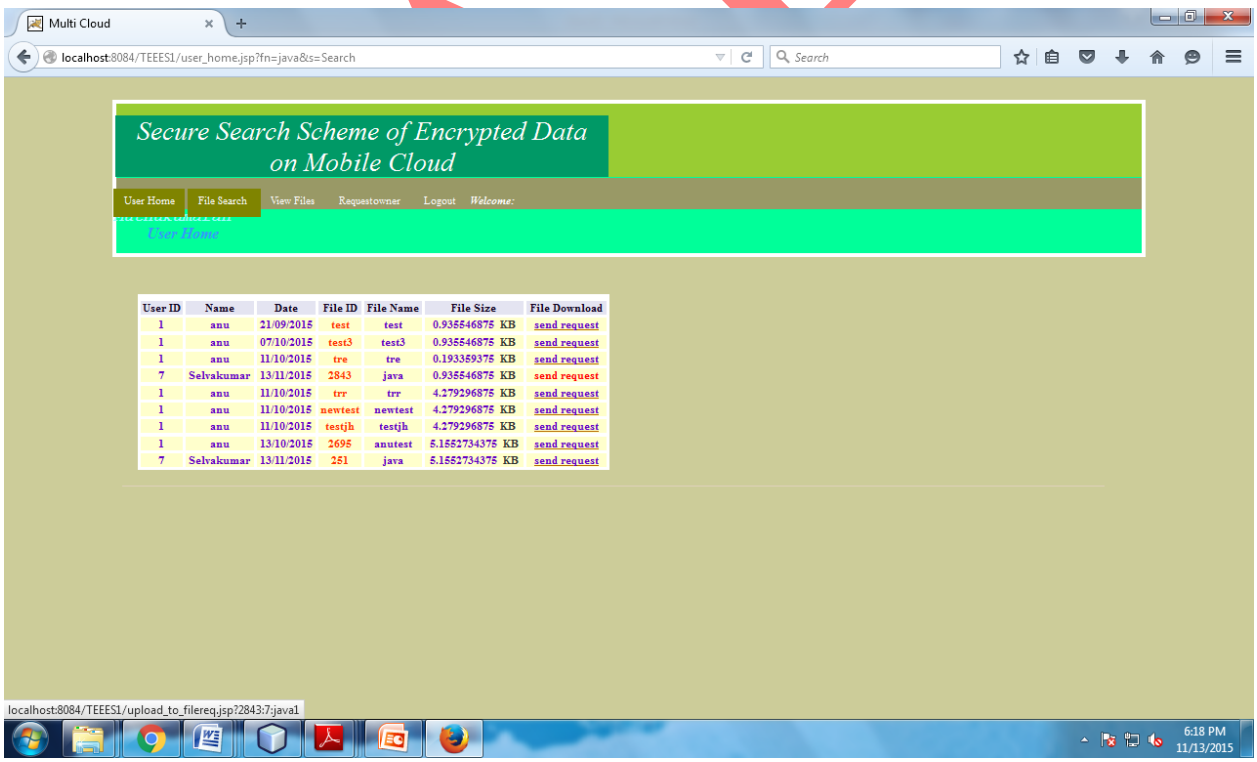

Fig 6 Search result and send request to corresponding user

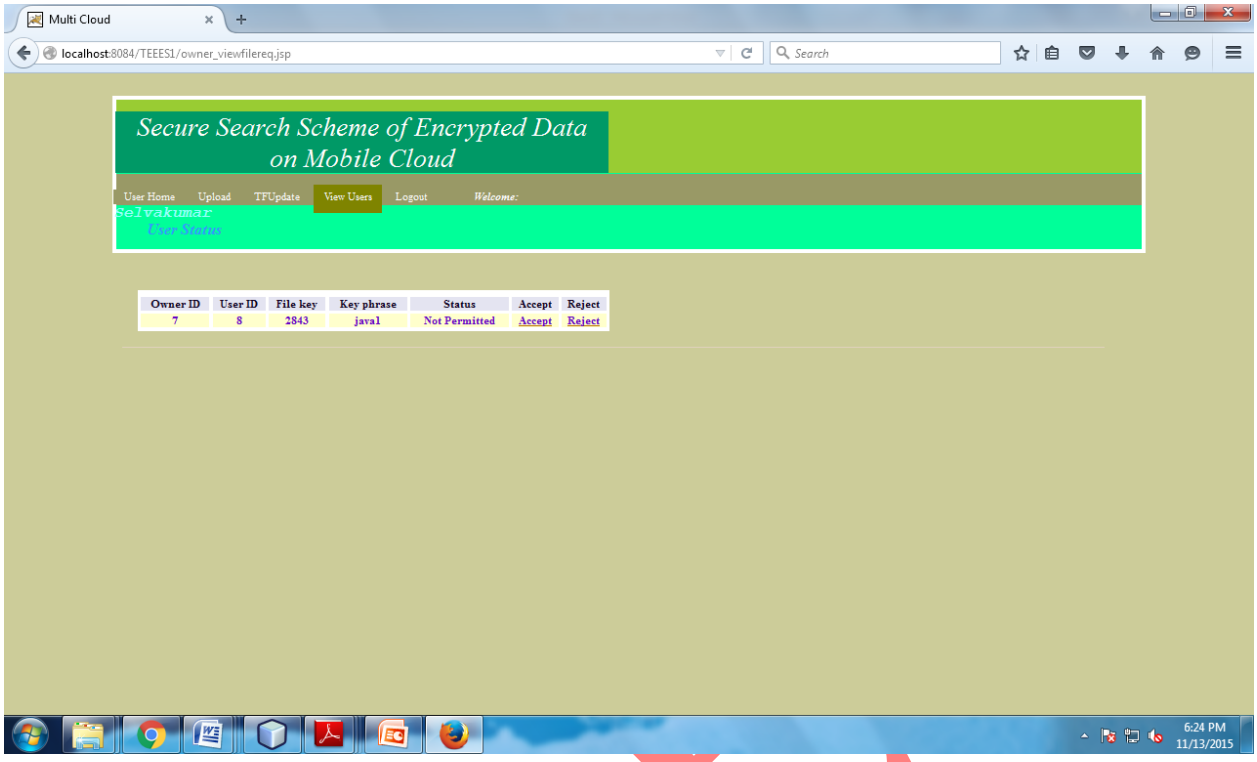

Fig 7 Data Owner Accept the request

Fig 8 Data user download the file from server

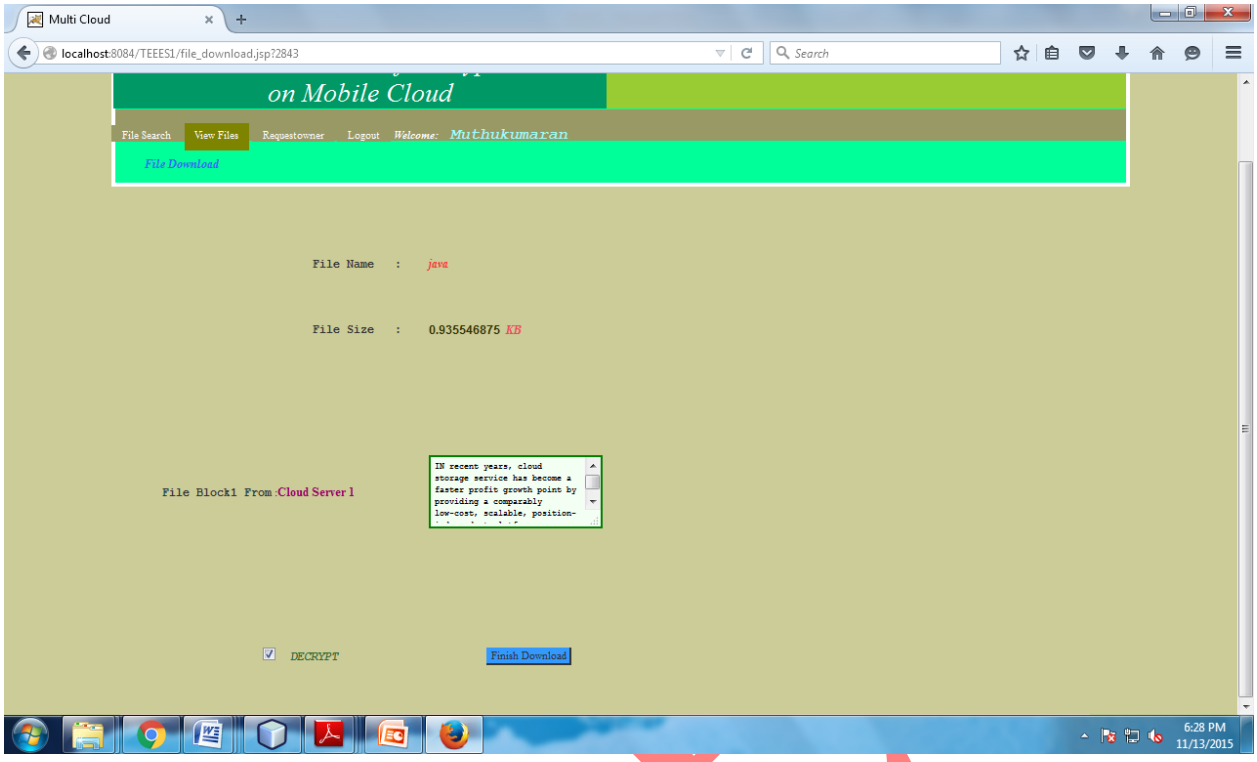

Fig 9 Data user Download the file

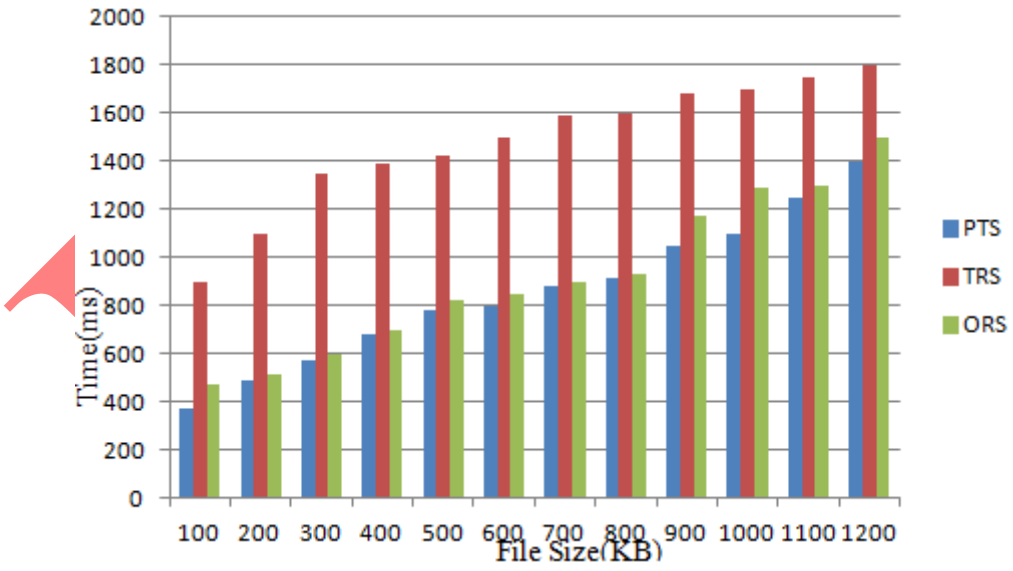

Fig 10 File search and retrival time

Fig 10 shows the graph of file search and retrieval time for different file sizes of different method. Among these methods PTS is the shortest since it does not have to perform any security computation. The FSRT of ORS is

effectively reduced when compared to the one of TRS. This difference is due to the advantages of the traffic and energy efficient encrypted keyword search design in terms of relevance score calculation offloading, and thus leads to reduction of file search and retrieval process.

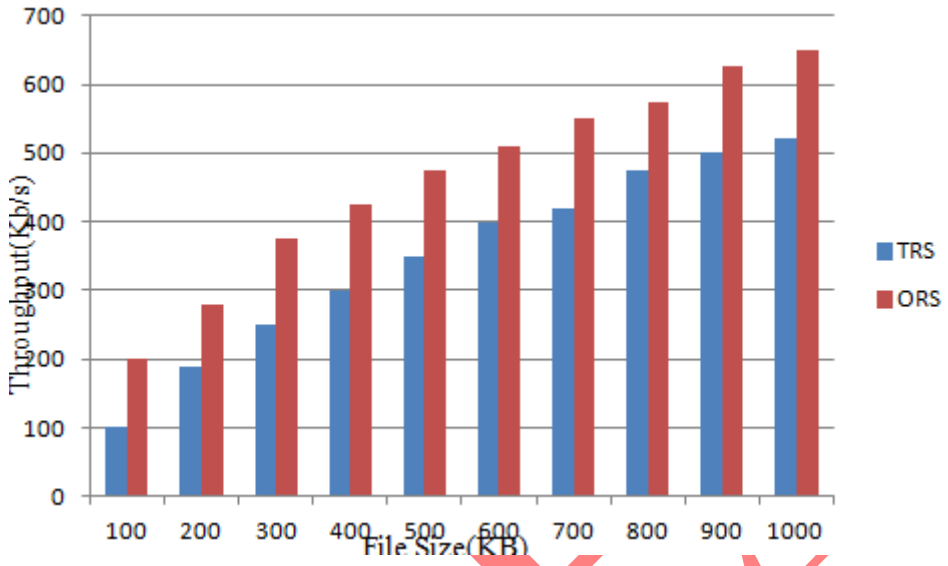

Fig 11 Throughput

Fig 11 shows the difference of file throughput of TRS and ORS. From this observation the file access acceleration is very effective when dealing with small files as the relevance score calculation is executed more frequently. For example, on a 100KB file, the access speed is increased from 104KB/s to 194KB/s, almost doubling the throughput.

V. CONCLUSION

In this project , as an initial attempt to create a traffic and energy efficient encrypted keyword search tool over mobile cloud storages. The security study of traffic and energy efficient encrypted keyword search showed that it is secure enough for mobile cloud computing, while a series of experiments highlighted its efficiency. Traffic and energy efficient encrypted keyword search is slightly more time and energy consuming than keyword search over plain-text, but at the same time it saves significant energy compared to traditional strategies featuring a similar security level.

REFERENCES

[1] Ankur Verma, Bhagyashri Patki, Priyanka Sawant, Prajkta Waingankar and Dike Onkar D (2015), “Efficient Similarity Search over Encrypted Data on Cloud”, International Journal of Current Engineering and Technology E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

[2] Antti P. Miettinen Jukka K. Nurminen (2010) “Energy efficiency of mobile clients in cloud computing”, dl.acm.org/citation.cfm?id=1863107

[3] Bing Wang, Shucheng Yuy, Wenjing Lou Y., Thomas Hou, (2014) "Privacy-Preserving Multi-Keyword Fuzzy Search over Encrypted Data in the Cloud" IEEE Transactions on cloud computing

[4] Cong Wang et al., (2012) "Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data", IEEE Transactions on parallel and distributed systems, vol. 23, no. 8.

[5] Cong Wang, Ning Cao, Jin Li, Kui Ren, and Wenjing Lou (2010) "Secure Ranked Keyword Search over Encrypted Cloud Data", dl.acm.org/citation.cfm?id=1846307

[6] Jianfeng Wang et al., (2013) "Efficient Verifiable Fuzzy Keyword Search over Encrypted Data in Cloud Computing" , Journal of Computer Science and Information system, volume 10, Issue 2.

[7] Jon Oberheide, Kaushik Veeraraghavan, Evan Cooke, Jason Flinn, Farnam Jahanian (2010) "Virtualized In-Cloud Security Services for Mobile Devices", https://jon.oberheide.org/files/mobivirt08-mobilecloud.pdf

[8] Li Chen, Xingming Sun, Zhihua Xiaand Qi Liu (2014) " An Efficient and Privacy-Preserving Semantic Multi-Keyword Ranked Search over Encrypted Cloud" Data International Journal of Security and Its Applications Vol.8, No.2, pp.323-332

[9] Ming Li et al., (2013) "Toward Privacy-Assured and Searchable Cloud Data Storage Services", IEEE Transactions on Network, volume 27, Issue 4.

[10] Zachary A. Kissel, and Jie Wang, (2013) "Verifiable Symmetric Searchable Encryption for Multiple Groups of Users", weblidi.info.unlp.edu.ar/WorldComp2013-Mirror/p2013/SAM9719.pdf