ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 11, November 2015**

# Privacy on Encrypted Cloud Data with Multi – Keyword Ranked Search

K.Balasubramani, M.Venkatesan

Assistant Professor, Dept. of CSE, TIET, Attur, Tamilnadu, India

M.E.,Student, Dept. of CSE, TIET, Attur, Tamilnadu, India

**ABSTRACT:** With the advent of cloud computing, data owners are motivated to outsource their complex data management systems from local sites to the commercial public cloud for great flexibility and economic savings. But for protecting data privacy, sensitive data have to be encrypted before outsourcing, which obsoletes traditional data utilization based on plaintext keyword search. Thus, enabling an encrypted cloud data search service is of paramount importance. Considering the large number of data users and documents in the cloud, it is necessary to allow multiple keywords in the search request and return documents in the order of their relevance to these keywords. Related works on searchable encryption focus on single keyword search or Boolean keyword search, and rarely sort the search results. In this paper, for the first time, we define and solve the challenging problem of privacy-preserving multi-keyword ranked search over encrypted data in cloud computing (MRSE). We establish a set of strict privacy requirements for such a secure cloud data utilization system. Among various multi-keyword semantics, we choose the efficient similarity measure of "coordinate matching," i.e., as many matches as possible, to capture the relevance of data documents to the search query. We further use "inner product similarity" to quantitatively evaluate such similarity measure. We first propose a basic idea for the MRSE based on secure inner product computation, and then give two significantly improved MRSE schemes to achieve various stringent privacy requirements in two different threat models. To improve search experience of the data search service, we further extend these two schemes to support more search semantics. Experiments on the real-world data set further show proposed schemes indeed introduce low overhead on computation and communication.

**KEYWORDS:** Cloud computing, MRSE; single keyword search, Boolean keyword search, coordinate matching, multi-keyword semantics, privacy-preserving

## I. INTRODUCTION

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services. These services typically provide access to advanced software applications and high-end networks of server computers. The goal of cloud computing is to apply traditional supercomputing, or high-performance computing power, normally used by military and research facilities, to perform tens of trillions of computations per second, in consumer-oriented applications such as financial portfolios, to deliver personalized information, to provide data storage or to power large, immersive computer games. The cloud computing uses networks of large groups of servers typically running low-cost consumer PC technology with specialized connections to spread data-processing chores across them. This shared IT infrastructure contains large pools of systems that are linked together. Often, virtualization techniques are used to maximize the power of cloud computing. A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider. Capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time. Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be managed, controlled, and reported providing transparency for both the provider and consumer of the utilized service

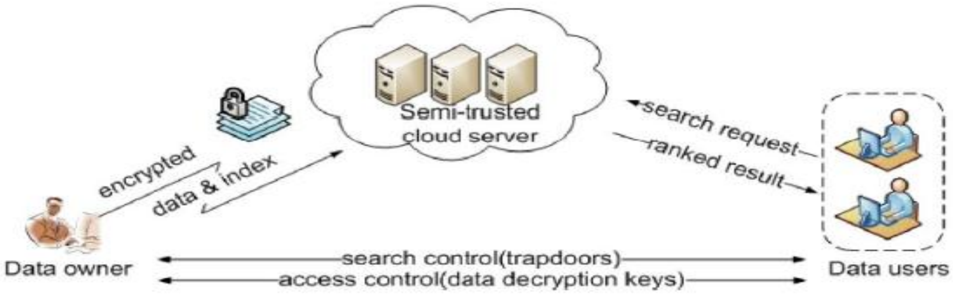

Semi-trusted cloud server
encrypted data & index
search request
ranked result
Data owner
Data users
search control(trapdoors)
access control(data decryption keys)

Fig1. Architecture of the search over encrypted cloud data

## II. .RELATED WORK

It is an important research problem to enable the cloud service provider to efficiently search for the keyword in encrypted files and provide user with efficient search result maintaining data privacy at the same time. We have researched on the following papers.

**Practical Technique for Search over Encrypted** Cloud Data This paper discusses on sequential scanning search technique [1] that searches over encrypted data stored in cloud without losing data confidentiality. The technique is provably secure and isolates the query result whereby the server doesn't know anything other the search result. It also supports functionalities such as controlled searching by server, hidden query support for user which searches for a word without revealing it to the server. With searchable symmetric encryption [7] and pseudorandom sequence generating mechanisms that are secure, encrypted data can be effectively scanned and searched without losing data privacy. The scheme that is proposed is flexible that it can be further extended to support search queries that are combined with Boolean operators, proximity queries, queries that contain regular expression, checking for keyword presence and so on. But, in case of large documents and scenarios that demand huge volumes of storage, the technique has high time complexity.

**Public Key Encryption with Keyword Search** Dan Boneh proposed a solution for searching over the cloud data that is encrypted using the Public key Crypto System [2]. The idea is to securely attach or tag the related keywords along with the each file. This will avoid the need to completely decrypt the file and save the time of scanning entire file to check if the keyword exists. The file is encrypted using a public key encryption algorithm [2] and containing keyword W, send only the Trapdoor (W) to server. He proposed two methods for construction of this scheme, one using the bilinear maps and other using Jacobi symbols. The problem with this scheme is that every tag of all the files has to be processed for finding the match. data hosting services in the context of Cloud Computing. 2.3 Boolean Symmetric Searchable Encryption Most of the techniques discussed so far focused only on single keyword matching but in real-time scenarios users may enter more than one word. Tarik Moataz came up with a solution to tackle such challenges of searching multiple keywords over the encrypted cloud data. The construction of Boolean Symmetric Searchable Encryption (BSSE) [11] is mainly based on the orthogonalization of the keyword field according to the Gram-Schmidt process. The basic Boolean operations are: the disjunction, the conjunction and the negation

The search query is also described as a binary vector where each bit means whether corresponding keyword appears in this search request, so the similarity could be exactly measured by the inner product of the query vector with the data vector. However, directly outsourcing the data vector or the query vector will violate the index privacy or the search privacy. To meet the challenge of supporting such multi keyword semantic without privacy breaches, we propose a basic idea for the MRSE using secure inner product computation, which is adapted from a secure k-nearest neighbor (kNN) technique , and then give two significantly improved MRSE schemes in a step-by-step manner to achieve various stringent privacy requirements.

## III. PROPOSED SYSTEM

### Data Owner Module

This module helps the owner to register those details and also include login details. This module helps the owner to upload his file with encryption using RSA algorithm. This ensures the files to be protected from unauthorized user.

DOI: 10.15680/IJIRCCE.2015. 0311220

**Data User Module**

This module includes the user registration login details. This module is used to help the client to search the file using the multiple key words concept and get the accurate result list based on the user query. The user is going to select the required file and register the user details and get activation code in mail email before enter the activation code. After user can download the Zip file and extract that file.

**Encryption Module:**

This module is used to help the server to encrypt the document using RSA Algorithm and to convert the encrypted document to the Zip file with activation code and then activation code send to the user for download.

**Rank Search Module**

These modules ensure the user to search the files that are searched frequently using rank search. This module allows the user to download the file using his secret key to decrypt the downloaded data. This module allows the Owner to view the uploaded files and downloaded files .

## IV. PROPOSED ALGORITHM

**RSA** is one of the first practical public-key cryptosystems and is widely used for secure data transmission. In such a cryptosystem, the encryption key is public and differs from the decryption key which is kept secret. In RSA, this asymmetry is based on the practical difficulty of factoring the product of two large prime numbers, the factoring problem. RSA is made of the initial letters of the surnames of Ron Rivest, Adi Shamir, and Leonard Adleman, who first publicly described the algorithm in 1977. Clifford Cocks, an English mathematician, had developed an equivalent system in 1973, but it was not declassified until 1997. A user of RSA creates and then publishes a public key based on two large prime numbers, along with an auxiliary value. The prime numbers must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, if the public key is large enough, only someone with knowledge of the prime numbers can feasibly decode the message. Breaking RSA encryption is known as the RSA problem, whether it is as hard as the factoring problem remains an open question.RSA involves a public key and a private key. The public key can be known by everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted in a reasonable amount of time using the private key. The keys for the RSA algorithm are generated the following way:

Choose two distinct prime numbers p and q.

- For security purposes, the integers $p$ and $q$ should be chosen at random, and should be of similar bit-length. Prime integers can be efficiently found using a primality test.
- Compute $n = pq$.
- $n$ is used as the modulus for both the public and private keys. Its length, usually expressed in bits, is the key length.
- Compute $\varphi(n) = \varphi(p)\varphi(q) = (p - 1)(q - 1) = n - (p + q - 1)$, where $\varphi$ is Euler's totient function.
- This value is kept private. Choose an integer $e$ such that $1 < e < \varphi(n)$ and $\gcd(e, \varphi(n)) = 1$; i.e., $e$ and $\varphi(n)$ are coprime.$e$ is released as the public key exponent’s having a short bit-length and small Hamming weight results in more efficient encryption – most commonly $2^{16} + 1 = 65{,}537$.
- However, much smaller values of $e$ (such as 3) have been shown to be less secure in some settings.[8]Determine $d$ as $d \equiv e^{-1} \pmod{\varphi(n)}$; i.e., $d$ is the modular multiplicative inverse of $e$ (modulo $\varphi(n)$).This is more clearly stated as: solve for $d$ given $d{\cdot}e \equiv 1 \pmod{\varphi(n)}$This is often computed using the extended Euclidean algorithm. Using the pseudo code in the Modular integers section, inputs $a$ and $n$ correspond to $e$ and $\varphi(n)$, respectively is kept as the private key exponent. The public key consists of the modulus $n$ and the public (or encryption) exponent $e$.
- The *private key* consists of the modulus $n$ and the private (or decryption) exponent $d$, which must be kept secret. $p$, $q$, and $\varphi(n)$ must also be kept secret because they can be used to calculate $d$.An alternative, used by PKCS#1, is to choose $d$ matching $de \equiv 1 \pmod{\lambda}$ with $\lambda = \mathrm{lcm}(p - 1, q - 1)$, where lcm is the least common multiple. Using $\lambda$ instead of $\varphi(n)$ allows more choices for $d$. $\lambda$ can also be defined using the Carmichael function, $\lambda(n)$.Since any common factors of (p-1) and (q-1) are present in the factorization of p*q-1,[9] it is recommended that (p-1) and (q-1) have only very small common factors, if any besides the necessary 2.[10] [11]

**Encryption**

Alice transmits her public key ($n$, $e$) to Bob and keeps the private key $d$ secret. Bob then wishes to send message $M$ to Alice. He first turns $M$ into an integer $m$, such that $0 \leq m < n$ and gcd($m$, $n$) = 1 by using an agreed-upon reversible protocol known as a padding scheme. He then computes the cipher text $c$ corresponding to

$c=m^e(\text{mod n})$

This can be done efficiently, even for 500-bit numbers, using Modular exponentiation. Bob then transmits $c$ to Alice. Note that at least nine values of $m$ will yield a cipher text $c$ equal to $m$,[note 1]

**Decryption**

Alice can recover $m$ from $c$ by using her private key exponent $d$ via computing

$m=c^d(\text{mod n})$

Given $m$, she can recover the original message $M$ by reversing the padding scheme.

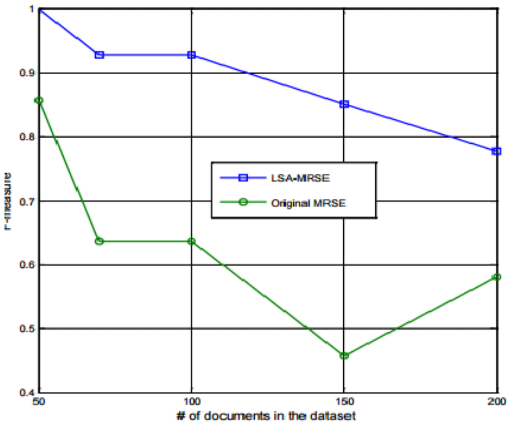
**Chinese remainder algorithm**

For efficiency many popular crypto libraries (like openSSl, Java and .NET) use the following optimization for decryption and signing based on the Chinese remainder theorem. The following values are precomputed and stored as part of the private key: and : the primes from the key generation,These values allow the recipient to compute the exponentiation $m = c^d$ (mod pq) more efficiently as follows:(if then some libraries compute h as )This is more efficient than computing $m \equiv c^d$ (mod pq) even though two modular exponentiations have to be computed. The reason is that these two modular exponentiations both use a smaller exponent and a smaller modulus.

**Performance Analysis** F-measure that combines precision and recall is the harmonic mean of precision and recall [8]. Here, we adopt F-measure to weigh the result of our experiments.

$$F = \frac{2.\text{precision}.\text{recall}}{precision + recall}$$

or a clear comparison, our proposed scheme attains score higher than the original MRSE in F-measure. Since the original scheme employs exact match, it must miss some similar words which is similar with the keywords. However, our scheme can make up for this disadvantage, and retrieve the most relevant



## V. SIMULATION RESULTS

Multiple users are created at a centralized location for the data owners and data users. We can see that either of the users can access the system once they login. The exchange of communication between data owners and data users is strictly through E-mail system which enables the system to be secured. Since the contents are encrypted and kept in the cloud, public viewing of these files is impossible. The files or contents can be viewed only after the consent of the data owners, after getting the secret key.

## VI. CONCLUSION AND FUTURE WORK

In this paper, for the first time we define and solve the problem of multi-keyword ranked search over encrypted cloud data, and establish a variety of privacy requirements. Among various multi-keyword semantics, we choose the efficient similarity measure of “coordinate matching,” i.e., as many matches as possible, to effectively capture 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, we propose a basic idea of MRSE using secure inner product computation. Then, we give two improved MRSE schemes to achieve various stringent privacy requirements in two different threat models. We also investigate some further enhancements of our ranked search mechanism, including supporting more search semantics, i.e., TF _ IDF, and dynamic data operations. 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. In our future work, we will explore checking the integrity of the rank order in the search result assuming the cloud server is untrusted.

## REFERENCES

1. N. Cao, S. Yu, Z. Yang, W. Lou, and Y. Hou, “LT Codes-Based Secure and Reliable Cloud Storage Service,” Proc. IEEE INFOCOM, pp. 693-701, 2012.
2. D. Song, D. Wagner, and A. Perrig, “Practical Techniques for Searches on Encrypted Data,” Proc. IEEE Symp. Security and Privacy, 2000.
3. 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.
4. R. Curtmola, J.A. Garay, S. Kamara, and R. Ostrovsky, “Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions,”Proc. 13th ACM Conf. Computer and Comm. Security (CCS ’06), 2006.
5. 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.

## BIOGRAPHY

M.Venkatesan M.E., Student, Dept.of CSE, Tagore Institute of Engineering and Technology, Attur, Tamilnadu, India

K.Balasubramani, M.E., Assistant Professor, Tagore Institute of Engineering and Technology, Attur, Tamilnadu, India